CONNEXIONS™

The Interoperability Report

October 1990 | *Special Issue: INTEROP 90 Backgrounder* | Volume 4, No. 10

ConneXions — The Interoperability Report tracks current and emerging standards and technologies within the computer and communications industry.

In this issue:

ConneXions is published monthly by Interop, Inc., 480 San Antonio Road, Suite 100, Mountain View, CA 94040, USA. 415-941-3399. Fax: 415-949-1779.

ISSN 0894–5926

From the Editor

Welcome to INTEROP 90! This special edition of *ConneXions*—over twice the size of a "normal" issue—is designed to complement the conference program with "background reading" on a few select topics. Needless to say, with 45 sessions and around 200 speakers, an issue containing articles on *every* topic would turn into a book. We will of course continue to cover conference topics in future issues, so I encourage you to sign up for a subscription now and ensure that you stay informed in this rapidly changing field.

The first 5 articles in this issue address emerging technologies which are being showcased at INTEROP 90; *Distributed Computing Environments, FDDI, ISDN, SMDS* and *The X Window System.* The articles are by Mike Millikin, Mark Wolter, Dory Leifer, Larry Hughes/Steve Starliper, and Wayne Dyksen/Tim Korb respectively. You'll see and hear a lot about these topics this week—in tutorials, conference sessions, informal BOF sessions, and through numerous special technology demonstrations on the exhibit floor. Check your conference program for further details.

Networks are currently being built using several forms of electromagnetic radiation in place of conventional wire. Richard Allen gives an overview of these technologies in an article on page 42.

Craig Partridge describes the state of very-high speed (gigabit) networking in a brief article starting on page 46.

INTEROP®90

Network management is a major topic at INTEROP 90. Reflecting this, Bruce Murrill presents a profile of the OSI/NM Forum. We will follow this next month with an article entitled "Network Management Directions" by Karl Auerbach and Denis Yaro.

"Multiprotocol Internetting" is the overall theme for INTEROP 90. We envision the future of networking as not necessarily being dominated by one protocol suite (such as OSI). Instead, a number of diverse technologies will need to coexist and interoperate. This is a major technical challenge, and it is addressed in several sessions. One existing architecture which will have to be part of this protocol spectrum is DECnet, or *DECnet/OSI Phase V.* Carl Malamud examines the future of DECnet and its relationship to OSI.

At the end of this issue you'll find a list of suggested articles for further reading.

Over the next several months, we will follow up with articles from other INTEROP sessions and BOFs. A list of some of these articles can be found on page 26. We hope you enjoy your stay in San Jose!

Distributed Computing Environments

Laying the foundation for the future of interoperability

by Michael D. Millikin, Seybold's Office Computing Group

"No matter where you go, there you are." —Buckaroo Banzai.

Introduction

We're at the advent of a new style of network-based computing that fundamentally will alter the design and implementation of information systems for the rest of the decade. Call it third-wave, fourth wave, new wave, distributed network, cooperative network, or client-server computing, this new *Distributed Computing Environment* (DCE) will become the foundation for intra- and inter-company interoperability. Distributed Computing Environments are based on the premise that "The Network is the Computer." Ideally, such environments provide transparent access to data and computation across collections of multi-vendor, heterogeneous systems. The strategic architectures of every major vendor are now based on some form of distributed computing environment. In the Spring, the *Open Software Foundation* (OSF) announced its technology selections for its Distributed Computing Environment. The vendor response to the OSF *Request for Technology* (RFT) was staggering: Fifty vendors submitted technologies (only 29 of those were OSF members).

Some of you who mentally categorize OSF as being on the losing end of a contest with *UNIX International* to deliver a UNIX operating system may be a trifle surprised at the organization's interest in distributed networks.

OSF has a good reason for pushing so hard on the Distributed Computing Environment: the number one concern of its membership is interoperability. The membership of OSF is driven in turn by the concerns of its users. OSF almost seems to be becoming a lens focusing various technology efforts that really require consensus if not standardization. Some of the areas OSF is tackling are kernel-specific, but others—such as the DCE work—are not.

We're not going to adduce the business needs that mandate the transition to distributed network computing yet another time here. Instead, we will accept those needs as given, and will flesh out a technology framework for the discussion of DCE trends and the evaluation of DCE platforms and products.

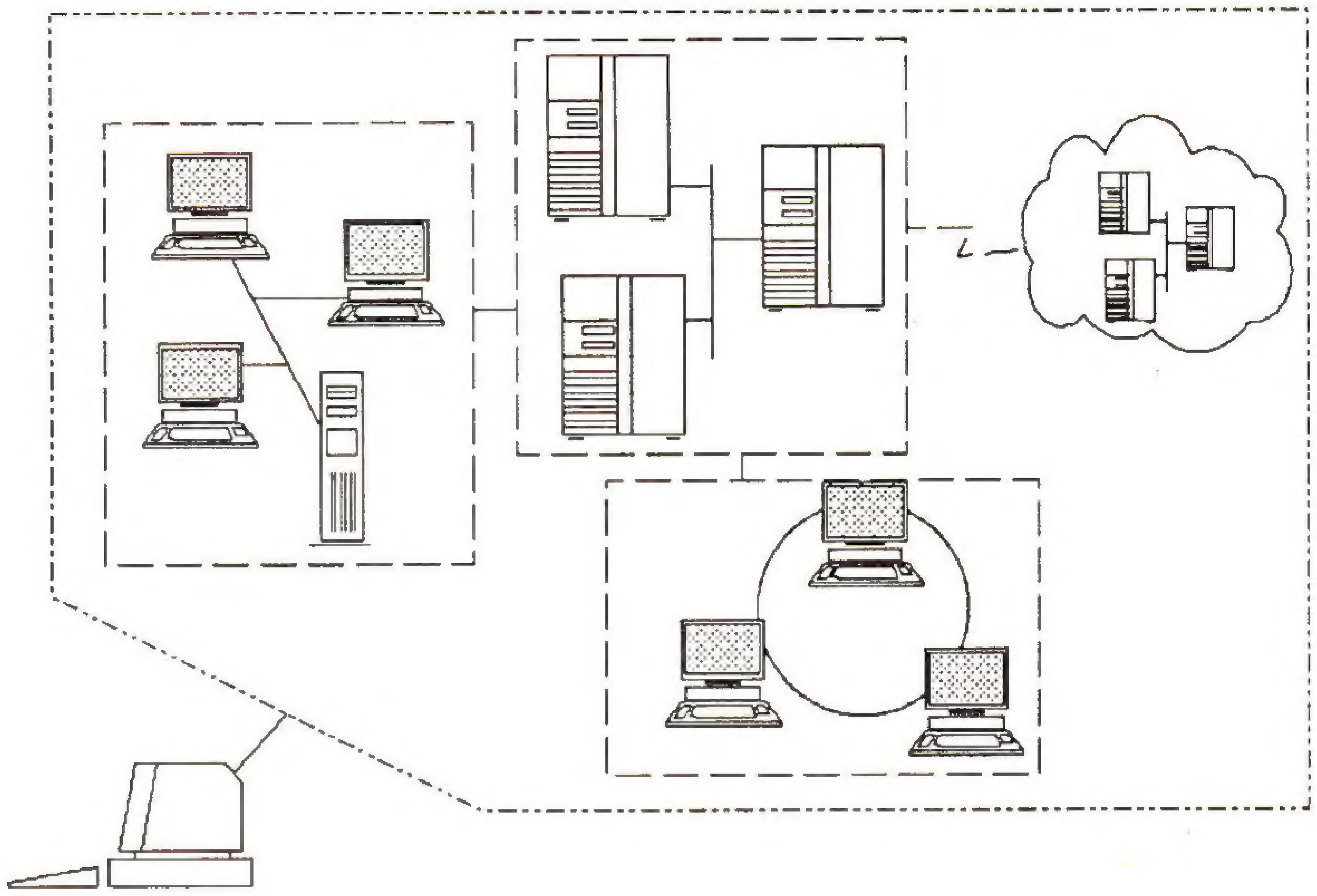
The OSF RFT has helped considerably in this process. OSF had originally hypothesized some 5 technology areas for DCE technology. The submissions actually encompassed 17 discrete areas, some of which OSF decided were out of the scope of its current evaluation project. Nevertheless, the surfacing of the technology framework provides a valuable reference point for discussion and extension.

What is a distributed computing environment?

The Distributed Computing Environment provides users and applications with transparent access to data, resources, and services strewn across a heterogeneous network. The key to realizing the theoretical benefit of such an architecture is *transparency*. Users can't spend their time thrashing about trying to figure out where something is. Nor should developers have to code into their applications locations for resources over the net. It is in no one's interest to force applications developers to become communications gurus. Nor should business users have to worry about mounting remote volumes. From the MIS viewpoint, the network should be manageable.

The final picture is one of a "virtual" network: a collection of workgroup, departmental, enterprise, and inter-enterprise LANs that appear as a seamless and easily accessed whole to the end user.

Such a network, which must be multivendor, masks the fundamental heterogeneity of its nature. In such a system, the network becomes the computer (a tag line several vendors explored using as a slogan until Sun trademarked it as "The Network is the Computer").



The virtual network delivers a single logical image of sprawling resources to the end user or to applications.

Within this schema, there is clearly a wide range of functionality and capability. Distributed computing can mean transparent access to a remote file. It can also mean an environment that lets me split up the processes of an application and parcel those processes out to available compute servers. The second scenario, which takes us closer to distributed operating systems, is further away from mainstream implementation than is the remote file system. But both are aspects of the same environment.

Architecture

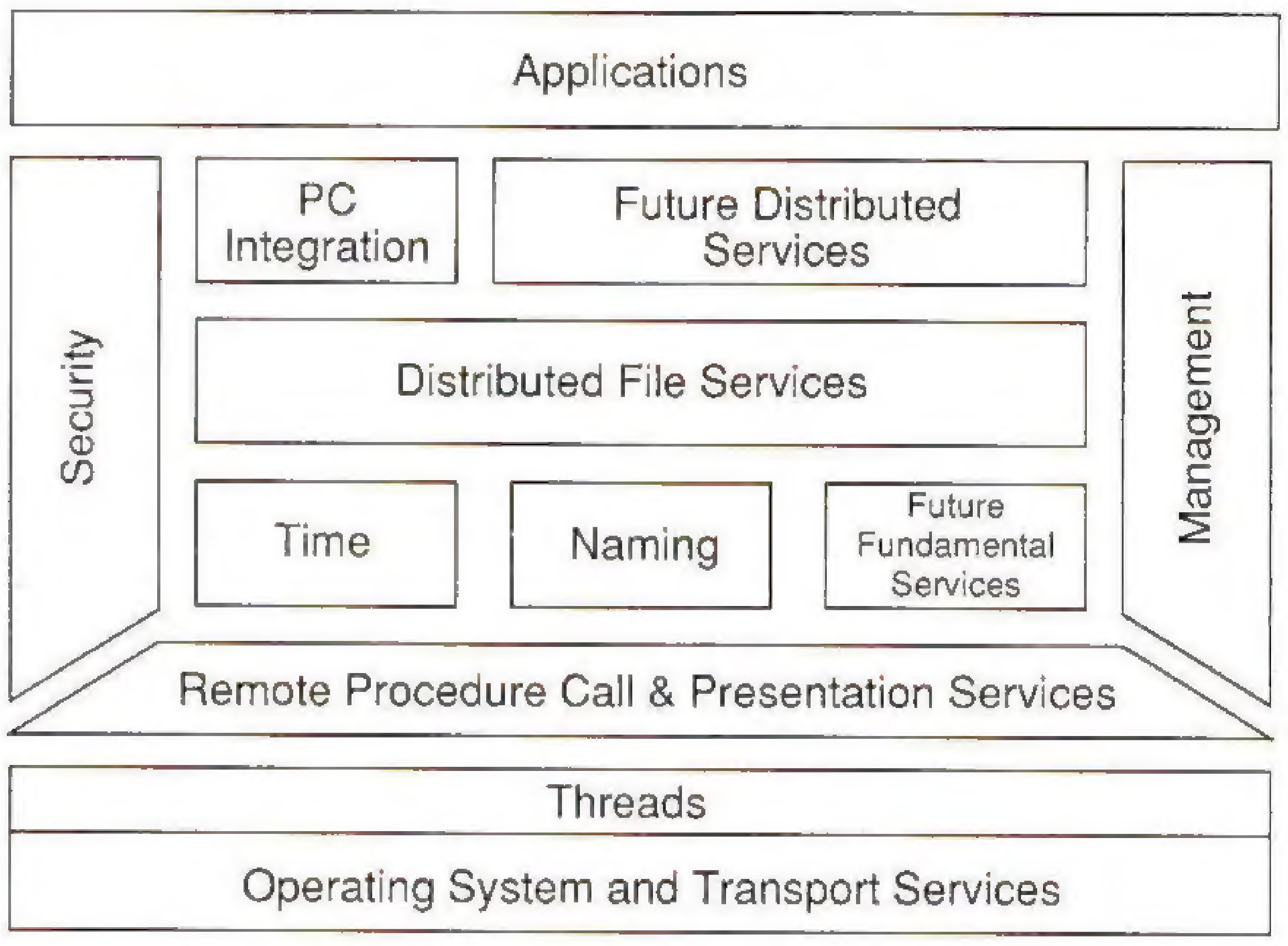
That, in turn, points out an important aspect of the emerging DCE—it should be architected in a rational manner to allow future extensibility. The degree of present architectural specificity is, however, currently a point of great contention. At the OSF members meeting discussing the DCE submissions, for example, one faction insisted that certain submissions were too grandiose, that we are too early in the process of the creation of DCE to have a soup-to-nuts plan. In rebuttal, another member offered the analogy of building a house. You could design and build a house one room at a time, this response went, but the result might look a little funny and might not produce an optimal building. On the other hand, drawing up a complete architecture from the beginning does not preclude the builder from building only one room at a time. The difference would be that the final structure would function much better as a building, rather than as a collection of tacked-together rooms. Of course, you could always rebut that by pointing out that even the architected building would probably need add-ons, and, by starting with a small number of rooms, you would be better able to create a unifying architecture at a later time when you knew what the additions would look like. And so on and so on, until you beat the analogy to death.

continued on next page

Distributed Computing Environments *(continued)*

Regardless of factional affiliation (architect-the-whole-now or architect-the-whole-later), those concerned with designing and implementing the DCE need to have a shared technology framework in order to discuss the relevance or timeliness of a given product or approach.

As noted above, we will use the OSF technology framework as a starting point for discussion. With its recently announced *Roadmap for Open Network Computing* (ONC), Sun is reinforcing the concept of the OSF architecture (although not the product selections). This piece is not a complete discussion of the relative merits of the technologies submitted to the OSF. Where we adduce specific products, it is to provide a fairly well-known example within a given category, or to highlight a particular approach.


Applications
Security
PC Integration
Future Distributed Services
Management
Distributed File Services
Time
Naming
Future Fundamental Services
Remote Procedure Call & Presentation Services
Threads
Operating System and Transport Services

The OSF architecture can serve as a blueprint for Distributed Computing

The DCE Technology Framework

The key characteristic of the emerging DCE is that it postulates the virtual extension of a compute systems over multiple systems distributed across a network. Using the "network is the computer" model, you quickly realize that, as a computer is more than a file system, a viable DCE must have more than a file system and a remote procedure call.

The needed components can range from system services such as security and file management to distributed file systems to distributed application processes. If the network is truly to be the computer, then all the aspects that make our computers into viable, functioning systems must be replicated and, in most cases, augmented, in order to provide transparency over a heterogeneous network.

It is important, particularly when examining areas of the distributed computing environment that generate a great deal of partisan enthusiasm, to remember that the DCE framework is multidimensional, and that technologies and technology areas are in many cases orthogonal. In other words, selecting a technology along one dimension does not directly affect a technology along another dimension. It is this ability to mix and match that OSF will exploit in its ultimate recommendation for a distributed computing environment.

Security

With the *Internet Worm* still fresh in many people's minds, security is a major issue within the Distributed Computing Environment. By its very nature, a distributed environment offers more points of entry to malefactor. Two major areas of concern within DCE security are *access control* and *authentication.*

Although many vendors have dealt with the issue of access control for an application on a homogeneous network, the problem obviously becomes more complex when dealing with a heterogeneous environment. In addition to the required security capabilities, access control in a DCE really requires some form of policy for security domains. In other words, as the scope of the problem expands, the capabilities of the applied solution need to become both more flexible and broader. Security domains, as a specific example, can reduce the administrative overhead required within a DCE.

Access control alone is only a partial solution, however. Although you can force a workstation or host to prove its identity, you can still end up relying upon the system's word as to the identity of a given user. Authentication services must exist within a distributed network environment where a workstation cannot be trusted to identify itself or its users correctly to shared network services. An authentication service is a mechanism for providing trusted third-party verification of user identities.

An authentication service, which basically requires the user to prove his or her identity for each required service, must be secure, reliable, transparent, and scalable.

Kerberos

A model for authentication services currently much in favor is *Kerberos,* which was developed as part of MIT's Project Athena. (Kerberos, for non-mytho-buffs, was the three-headed dog that guarded the entrance to the infernal regions. An interesting view of the nature of data servers, no?)

Kerberos uses *private key encryption:* Each Kerberos principal has a private key known only to the principal and to Kerberos. Kerberos maintains a database of clients and their keys. Although a network may run Kerberos on more than one machine, only one definitive copy of the Kerberos database exists. Other Kerberos servers may use read-only copies of the database to eliminate single point-of-failure issues.

Kerberos provides three levels of protection. The lowest requires only that authenticity be established at the initiation of a connection, assuming that subsequent network messages flow from the authenticated principal. The next level up requires the authentication of each network message. On the level beyond these safe messages are private messages, where each message is encrypted as well as authenticated. Both Sun and OSF have indicated their intention to use aspects of Kerberos in a security service. Additionally, Sun plans to continue to support and to enhance its *Secure RPC.*

Naming and Directory services

Once disparate systems are internetworked, issues such as global naming and directory services become fundamental to the seamless implementation of a distributed environment. Allowing users to find other users for mail messages is the least of the directory services' function. Application clients need to find servers, accounting and administration information needs to be built on top of the basic directory, and, ultimately, objects must be able to send messages to other objects. A naming and directory service, in short, enables *location independence* in the distributed environment.

The directory service must support large-scale and long-lived networks, address delegations of authority, support the use of aliases, and provide lookup and browsing mechanisms.

continued on next page

Distributed Computing Environments *(continued)*

A hierarchical directory service that supports both partitioning and replication also seems appropriate for supporting an effective distributed environment, especially since that is the structural model of the emerging X.500 directory standard.

The designers of X.500 intend the directory to support simple lookup, browsing, a yellow pages searching function, group membership and authentication services in a heterogeneous environment. X.500-based services seem to be the way the industry will go as quickly as possible, at least for a global directory service.

Since X.500 is still in the *Draft International Standard* (DIS) stage, many in the industry are making do in the interim with naming and directory services that position themselves closely to the emerging specifications. This two-tier approach, reflected in the OSF selections of *DECdns* and Siemens *DIR-X* for its naming service, will probably be a fairly common solution for the short to medium term.

The *DNA Naming Service* (DNS) from DEC provides a single global hierarchical namespace (replacing the node databases of Phase IV), and maps unique names to corresponding node addressed. DECdns stores object and resource names and attributes. Nodes keep a cache with current names. The namespace is stored in a partitioned, partially replicated database. Parts of the namespace may reside in different physical locations, and parts of the namespace maybe replicated in multiple physical locations.

DECdns offers caching, replication, and security capabilities. The caching and replication can handle very large systems; this was to be Digital's global name service for DECnet (with appropriate migration to X.500). The name service thus has the capability of supporting global WANs in the absence of yet unspecified X.500 capabilities. Furthermore, because of Digital's intention to incorporate X.500 specification at some point, the DECdns architecture has a great deal in common with that of X.500. DECdns also brings with it a set of management tools for the name service.

Sun plans to enhance its Naming Service by providing stronger consistency, improved security and a distributed binding service. Sun also plans to add features such as a global namespace for all network resources and provide a distribution strategy.

Time services

With so many different systems being tied together to act as one computer, all the clocks on these different systems should be synchronized. This is important for distributed file access as well as for distributed procedures and processes. For example, one system may be dependent on a piece of information from another remote system before a key request can be handled. Systems need to know when backups can be performed and how their downtime affects the other processes in the virtual network.

A distributed time service is one of those easily overlooked functions that is essential to the continued reliability and integrity of a distributed environment. Digital, again, has a nice little piece of technology to meet this need, the *Digital Distributed Time Synchronization Service* (DECdts). DECdts uses a different time synchronization algorithm than does the Internet counterpart, NTP *(Network Time Protocol)*. Both appear equally reliable. OSF is opting for DECdts; Sun plans to implement NTP.

Distributed file systems

Say "distributed networking" and the first thing most people think of is a distributed file system—probably Sun's *Network File System* (NFS). Since the files that users would like to access are stored throughout the virtual network, a distributed file system to manage and transparently locate these pieces of information is needed.

Sun designed NFS to provide interoperability among different machines and systems with performance comparable to a small local disk. Sun believed that a client or server crash should not affect other nodes on the network, that the file system should remain transparent to existing applications, and that administering the file system should be no more taxing than managing a local hard disk.

NFS has succeeded greatly at one level: It is widely licensed and has become recognized as a de facto standard. With its stateless protocol design, NFS allows clients to survive a server crash and a planned or unplanned shutdown. However, NFS really needs better cache consistency and locking for shared read/write. And, while the administration is no tougher than managing a local hard disk, that's still pretty bad—certainly not a task for the average user.

AFS

Furthermore, NFS was designed as a LAN-based distributed file system. To better address the expanded requirements of a WAN distributed file system, OSF is using the *Andrew File System* (AFS) from Transarc. Transarc was formed by a group from CMU to commercialize aspects of the Andrew System.

The AFS uses a global file space, allowing all clients to see the same file name space, regardless of location. By contrast, the NFS client refers to a subtree offered by a specific node. AFS also uses Kerberos authentication to manage autonomous administrative domains, analogous to independent Kerberos realms. The use of these autonomous administrative domains, or *cells,* endows AFS with excellent scalability. The cells cooperate to provide the global file space.

AFS tries to minimize network and server loads by caching accessed data on the local disk and caching the status of the data in memory. In AFS, the server notifies the client in case the data at the server changes. This guaranteed callback ensures cache consistency. In Carnegie Mellon University's experience with AFS, the caching with callbacks allowed servers to handle 200 to 300 clients without performance degradation. (The CMU network is the largest AFS installation, with 3 cells, 25 servers, 1,200 clients, and 9,000 users.)

Sun plans to enhance NFS (in version 3) to include many (if not all) of the features offered by AFS. To improve performance, Sun plans to support local disk caching, and to tune the existing implementation of NFS. For security, Sun will use either Kerberos or Sun's own Secure RPC authentication services. Replication (a longer-term planned enhancement) will make NFS resources more available to users. And Sun plans to offer simpler commands and graphical tools to make the networks easier to configure.

There have been criticisms of networked file systems as being unreasonably slow. Some recent research is pointing out, however, that networks can actually achieve better performance with a high-performance server and a distributed file system than with slower local disks on workstations. (Coming up with optimal configuration for a given application need should provide vendors with plenty of opportunity to position their products against one another.)

continued on next page

Distributed Computing Environments *(continued)*

Distributed print service

A problem presented by a distributed heterogeneous environment is that since an application can reside anywhere and be accessed from anywhere, printing services have to accommodate this model. The notion of a printer being controlled and accessed by a single system no longer applies. Printing has to migrate from the hierarchical model to the notion of a distributed printing service available to all nodes in the distributed computing environment.

There are several interesting models floating around, such as the Palladium print service software from MIT's Project Athena. (MIT is submitting a number of major network services to OSF: Kerberos; Hesiod, naming; Zephyr, notification; Moira, service management; and Palladium.)

Terminal service

In the future, the prices will drop low enough to permit more users to have workstations on their desks. There will still be a role for low-cost terminals on the desktop, however. Therefore, even within DCE, accommodation must be made for terminals. Support for terminals could range from support for the newer X-terminals to a virtual terminal service for older, character-based tubes.

PC interfaces

Because so much critical data resides on PCs, support for DOS is an issue of importance. At a low level, PC NFS is an example of a solution for client file system access. However, it's clear that PCs need to play a much broader role within DCE simply because they are so pervasive. Also, as more powerful PCs (UNIX or OS/2 clients, or later Macs) earn more desktop space, the difference between the PC and the workstation erodes.

The two most likely solutions for integrating PCs into the DCE framework also correspond to the PC LAN operating system platforms with the greatest presence: *NetWare* and *LAN Manager*.

Currently, the relative share of these two is a bit skewed. NetWare is by far the heavyweight. However, LAN Manager should be picking up steam in the market for a variety of reasons:

- The endorsement and subsequent implementation of LAN Manager by systems vendors such as Digital, HP and IBM.
- The multiprocessing implementations of LAN Manager.
- The delivery of LAN Manager for UNIX (LM/X).

We are seeing the creation of two primary axes within the evolution of the distributed environment. One is an extension of the old Apollo *Network Computing System* (NCS) camp, represented now mainly by HP, Digital, and IBM (with alliances with smaller, strategic vendors, including Microsoft). The other is the old Sun *Open Network Computing* (ONC) camp, represented by Sun and the UNIX International licensees. The dynamic of competition and collaboration between these two groups is affecting most areas in DCE, and we will touch upon them a bit later.

With respect to PC interfaces, however, it is interesting to note that LAN Manager has also been submitted for consideration to OSF as part of a very comprehensive, partnered submission from HP, Digital, IBM, Microsoft, Transarc, and Locus.

Perhaps the closest thing to a submission from the other camp in this area is Netwise with its RPC compiler technology. Following on the heels of the Sun/Netwise/Novell agreement on RPC tools, accepting the Netwise technology would be tantamount to providing a fairly straightforward avenue for the integration of NetWare LANs into the distributed environment.

Because of the heat generated by any discussion of the PC LAN area, we expect this particular issue to become quite contentious. Netwise can point to its support of NetWare, coming support for Sun and original support for LAN Manager as justification for its being considered as a de facto industry presence. On the other hand, the Netwise technology doesn't fit neatly into the plans of the NCS/LAN Manager group. User needs are going to drive an accommodation of both, it seems. HP, for example, although one of the principal OSF members and with a strategic stake in LAN Manager/X, is also supporting NetWare as a LAN platform for its office software solutions.

IPC, RPC, and Presentation Services

Communications between applications and between remote systems is a key component that enables the creation of the distributed computing environment. At the first OSF members meeting concerning the DCE RFT, there was remarkable unanimity of mind in opting for a *Remote Procedure Call* (RPC) mechanism as the primary enabler of interoperability across heterogeneous systems.

The RPC extends the procedure call mechanism familiar to programmers for transferring control and data within a program across a network. Remote or distributed parts of the application thus simply become procedures that execute on a remote machine. The RPC uses a request/reply model of communication.

The requesting client issues a call to a local stub routine that handles all the network interactions necessary to complete the call and communicates with a counterpart stub on the server. The server stub then in turn deals with the server program, and communicates the result back to the client stub—and thence to the original requesting client. Through such mechanisms, the RPC shields the programmer from the underlying network.

Now there follows the politically charged discussion over which RPC to use. Basically, the two primary options facing the community are the Sun RPC from Open Network Computing (ONC), and the NCS RPC from the HP/Apollo *Network Computing Architecture* (NCA) with enhancements by Digital. (This isn't to say that there are no other RPC mechanisms available. But the two most visible in the market are the ONC and NCA variants.)

Although nominally the two RPC mechanisms perform comparable functions, they are actually quite different architecturally. These architectural difference dampened the hopes of those who wished for the OSF to opt for a politically sanitized solution—trying to mix in the Sun RPC with the NCS architecture, for example.

Data representation

The major differences between the NCA RPC and the ONC RPC appear to be in their approaches to data representation, and their binding models. There used to be a difference in the approach to the interface language and in the approach to the underlying network mechanisms as well. Recent activity on Sun's part has reduced those differences, however.

continued on next page

Distributed Computing Environments *(continued)*

Sun's earlier RPC language and *rpcgen* tool proved a poor match for the capabilities of the NCA NIDL (*Network Interface Definition Language*). Pursuant to the relationship with Netwise (which has several former Apollo folks) Sun now will be offering a more robust, generalized interface language that offers developers access to a much broader market than just Sun workstations. Furthermore, the use of the *Transport Layer Interface* (TLI) in System V.4 is freeing Sun from a previous IP-orientation in the application programming interface.

There currently is a clear difference in the two models of data representation. When on a network of heterogeneous machines, you need some form of data representation protocol to take data from one type of system to another. The data representation protocol defines a way to present data so that machines with varying local representation can communicate typed values to one another. (As an example, VAXen represent integers with the least significant byte at the low address, while the 68000 family represents integers with the most significant byte at the low address).

XDR

Sun chose to use a canonical form of data representation called XDR (*eXternal Data Representation*). In the Sun scheme, all data is converted to the XDR canon, even if the data happens to be passing between two machines of the same type.

NDR

The NCA advocates pounced on this as waste. The *Network Data Representation* (NDR) uses a "receiver make it right" approach. In other words, it is up to the receiver to convert the data representation only when the incoming format differs from the receiver's native format.

ASN.1

Another benefit Sun gained from the recent Netwise agreement is a migration path to ASN.1 (*Abstract Syntax Notation One*), the ISO data representation standard. The Netwise RPCTOOL compiler produces ASN.1 encoding. NCS version 2.0 will support some mechanism for providing ASN.1 encoding as well. And in terms of data representation, it's important to note that OSI adopted a negotiated form of representation between client and server. In other words, the direction OSI is taking is not that of a strict canonical representation at all times.

Binding

Binding refers to the process of establishing a reference to the target of a remote call—a major issue for developers who need to locate the services desired for their distributed applications across the network.

NCA uses an object-oriented binding model. In other words, it encourages developers to think of remote calls as operations upon objects, not as calls to specific machines or server processes. The goal of the NCA designers was to reduce the precision with which a developer must specify the location of a target for a call.

A very important aspect of the NCA object-oriented model is the use of a 128 bit, fixed-length *Universal Unique Identifier* (UUID) as the lowest level identification mechanism for any entity on the network. The NCA architects claim that the advantages of the UUID over a simple string identifier at the lowest level of the system include the smaller size, the ease of embedding the UUID in data structures, location transparency, and the ability to layer various naming strategies on top of the primitive naming mechanism.

NCA is architected to exploit the capabilities of all its components. Thus, the NCA/RPC packet format has a space for an object UUID. And the NCA-defined interfaces for talking to the Location Broker (a distributed application that aids client applications in finding objects) also have arguments that are object UUIDs. (For reasons like this, the concept of swapping out the NCA/RPC for another (such as the Sun RPC) as a political expediency is anathema to the NCA camp, and rightly so. The result would be worse than a camel.)

Location Broker

Within the NCA schema, a third-party becomes involved in the client/server communication: a broker. The client uses a broadcast binding to find the *Location Broker* (LB). The Location Broker then returns the network address of a server process that is willing to handle operations for a specified client object.

Developers not wishing to become embroiled with the object-oriented capabilities of NCA can bypass those object-oriented features and can use NCA as a conventional RPC system. (The benefit here, presumably, being the potential for migration in the future to the object-oriented model.)

In the ONC model, the client application first must establish contact with the server through an RPC creation procedure that specifies the name of the host on which the server process is running, the name and version number of the program to be called, and the network name used to reach the server. A special RPC service, *rpcbind,* locates the server program and the listener, or *inetd,* procedures to start the server if that application isn't running. Another procedure then returns the client handle.

A special broadcast service within *rpcbind* can call all servers of a particular type on a LAN without knowing the hosts on which those servers run. The *rpcbind* broadcast sends a message to all *rpcbind* servers requesting that they in turn relay the call to any server running on their machines. It is this mechanism to which Sun proponents point when they claim that ONC has as much capability for providing location services as does NCA with LB. And, indeed, this *rpcbind* service does undermine the NCA claim that ONC binding calls must all be explicit. However, even Sun and AT&T point out that this broadcast mechanism is heavy on network resource usage.

Distributed development environments

One of the major gating factors blocking widespread implementation of DCE is the availability and sophistication of developers' tools. Requiring application developers to become communications gurus is counterproductive. (Some might call it a brain-dead approach.) The DCE must offer transparency to the developer as well as to the end user.

There are a variety of levels where developers' tools must appear. Initially, the market has a requirement for a transport and RPC-independent programmatic interface that will protect developers from writing to a host of different transports and RPCs.

A future requirement will be for tools that help the developer construct distributed applications, including those that will split an application in the most effective manner. Also needed are methodologies for writing applications for DCE. System criteria such as security and scalability have to be taken into account in the development environment.

continued on next page

Distributed Computing Environments *(continued)*

The scope of the development environment must go beyond the application issues themselves. Security is a concern (enter services such as Kerberos). But there is another need for tools that will help developers measure and evaluate configuration and performance when designing distributed applications. This will help developers design applications which take advantage of the network and which are optimized for that environment.

Task Broker

Providing development tools for the DCE will become an area where vendors strive to distinguish themselves competitively, we believe. HP has already taken an interesting approach to this with its *Team Computing* program. Ultimately, Team Computing will define a family of products that incorporate some of the advanced distributed network computing technology of NCS. The first deliverable out of the Team Computing program, the *Task Broker,* has nothing to do with NCS, however. It is basically an intelligent batch facility that parcels out entire applications to available and appropriate compute nodes on the network. There is no need to change existing application code. The Task Broker provides a very easy point of entry for distributed network computing, and allows users to get an immediate appreciation of the benefits of such an environment. So attractive has this approach been that several large customers have requested alpha code, rather than the full-cycle QAd deliverable. (Developers at HP labs actually came up with Task Broker for their own use.)

Currently, there are two main contenders for basic development tools for the distributed computing environment. One is the Netwise RPC-TOOL compiler; the other is the *Network Interface Definition Language* (NIDL) within NCS. Each requires the developer to write only for a given set of APIs and then generate the appropriate stub code transparently. There are other mechanisms for generating such code, such as *rpcgen* within Sun's Open Network Computing environment, but these don't have the potential for providing an acceptable solution to the market in heterogeneous environments.

Migration and conversion aids

Most end users will not move from their traditional computing environments to DCE overnight. They will, instead, migrate slowly to integrate their existing applications and systems. Therefore, users need to understand what it takes to make the migration. They also require software conversion aids that help them migrate a conventionally written application to DCE. At the same time, they need tools that will allow for part of the system to be based on DCE while the rest is still based on the older hierarchical computing model.

Transaction services

For a DCE to be viable commercially, it must support *On-Line Transaction Processing* (OLTP). These users need the confidence of knowing that transaction integrity is guaranteed. In the long term, this should extend to the heterogeneous database environment.

Distributed operating systems

In the long term, users will gain better utilization of their available resources if processes and the operating system itself can be migrated or distributed across a network.

There is a difference between distributing procedures and distributed processes. The procedure, using a remote procedure call, is a concept easily adopted by most programmers. A process represents a finer degree of granularity within the application, however.

Distributing those finer-grained processes of an application out over a network requires either a great deal of expertise and experience or a powerful tool set. Ultimately, however, the distribution of application and system processes across multiple processors or multicomputers within a network will give us even greater power at our disposal.

Within the scope of its RFT for the DCE, OSF received a variety of responses with differing capabilities, ranging from the *Transparent Computing Facility* (TCF) from IBM and Locus to the *Chorus/MIX* distributed operating system from Chorus Systems in France.

Chorus

Chorus provides a network-transparent IPC facility (low-level IPC and RPC) as a foundation to support standard, transparently distributed, open operating systems services such as UNIX services in Chorus/MIX. Chorus has three main characteristics:

- A minimal kernel that can underlie a variety of operating systems, providing distributed processing and communication.
- Real-time services with multithreading and preemptive, fixed-priority scheduling.
- A machine-independent, modular architecture that supports parallel and multiprocessing systems.

The Chorus nucleus has both local and global responsibilities. On a local level, it has three components:

- *Real-time Executive,* which provides preemptive, fixed-priority scheduling and synchronization.
- *Virtual Memory Manager* (VM Manager), which directs local memory allocation and structures virtual memory address spaces.
- *Hardware Supervisor,* which provides dynamic loading of external events (such as interrupts, traps, and exceptions).

On a global level, there is the *IPC Manager,* which issues messages transparently to any node in the system. The IPC Manager keeps track of where messages are going and delivers them via RPCs and asynchronous message passes. Sometimes, external system servers are called upon to support various network protocols. The Chorus nucleus structure hides its distributed nature from those using it. Local services compute locally, and global services rely on the cooperation among nuclei to cope with distribution. IPC is the only communication tool; all sites use IPC rather than dedicated protocols.

What do you get out of such distributed operating system design (along with that of Amoeba, Mach, and Stanford's V)? High performance and utilization of multiprocessing and multicomputers. Distributed (and object-oriented) operating system technology will play an increasingly important role in years to come in the realization of distributed computing environments.

As we gain the capability of transparently distributing processes across commercially available networks, we come closer to the time when parallel processing and neural networks will become powerful and common technologies. In research labs, developers currently are at work on such multicomputers that rely upon proprietary communications schemes to achieve high-performance distribution of processes. Such capabilities will eventually appear on commercial networks, given the proper technology foundations. None of these technologies made OSF's initial cut at the DCE.

continued on next page

Distributed Computing Environments *(continued)*

Distributed system (network) administration and management

Network management and administration is a key concern within the homogeneous systems environment. There is a need to know what is happening within the network to make sure the network is being run efficiently and effectively and to handle and resolve problems. The same management issues surface in a heterogeneous environment. However, because in a heterogeneous environment a user is dealing with so many more systems, applications, and data, the problems and requirements are magnified.

Although some aspects of this problem are being addressed by certain submissions in response to the DCE RFT, OSF has recently released its RFT for a *Distributed Management Environment* (DME).

License server

Software distribution becomes a major issue in a distributed environment, especially when parts of applications and operating systems are distributed. Therefore, there needs to be a method for providing users on different systems with updated code. The *license server* handles this problem by allowing servers to distribute appropriate software to all nodes.

A license service also controls the number of users accessing a piece of software so that a vendor can keep track of license fees. In addition, it is a means of ensuring that all users are on the same software releases, and it can be used to update distributed directory services. Safeguards must be built into the license service concept. For example, there may be instances when users are not ready to update software. Therefore, they need the option of upgrading software on a delayed basis. And finally, license server technology can provide a low-risk way for users to try new software, which could be loaded on a server for demo purposes.

Extensibility

No single vendor can expect to control the whole computing environment of an end user. There are too many variables and specialized needs. Therefore, the environment should support extension by third parties.

Ruminations

We have taken a very quick trot through the major technology areas of distributed network computing—somewhat the magazine analog to the forced itinerary vacation where the tour guide hustles you off the bus, troops you through elegant buildings and then stuffs you back onto the bus to roar off to the next destination. We apologize, but we believe that the high-level view given in this write-up is valuable as a starting point.

The marketing rhetoric catalyzed by the OSF decision in May should have subsided a bit by now, allowing users to begin evaluating which set of technologies meets their particular needs.

Also percolating in the background of all of this is the ISO work on *Open Distributed Processing* (ODP). We expect, though, that much of the ongoing work by vendors submitting to the OSF will influence the ODP work in ISO (as some ISO work already has influenced the OSF submissions).

There is not an easy or simple solution to resolve the different proposals from the various camps involved in the process. There are installed bases to consider, existing technologies and implementation that would have to be accommodated, and compromises to be made in order to provide mutual benefit.

The RFT process has done the industry a great service by opening up the perspectives of those engineers who may have held a more parochial view of their position in the industry. We are not technology poor. We don't have to slow our progress to wait for the proper vehicles to speed us along. Much is already there.

DCE function	***OSF DCE***	***Sun ONC***
RPC, presentation	HP, Digital NCS, NDR	Sun RPC, XDR
Network Compiler:	HP, Digital NIDL	Netwise RPCTool
Distributed File System:	Transarc AFS	Sun NFS
Distributed Time Service:	Digital DECdts	Internet NTP
Directory Service:	Digital DECdns, Siemens DIR-X	Sun NIS
Security Service:	MIT Kerberos	MIT Kerberos, Sun Secure RPC
Threads:	DEC CMA	Sun RPC
PC Integration:	LM/X, Sun PC-NFS	Sun PC-NFS

Technology choices in Distributed Computing Environments

MICHAEL D. MILLIKIN is Vice President and Senior Analyst for Patricia Seybold's Office Computing Group (PSOCG). The Group is an industry publishing, research, and consulting company. It provides analysis, education and direction concerning the development of technology and its relationship to organizational performance. Mr. Millikin specializes in following the emerging distributed network computing architectures and the application sets built atop them. He consults to both the vendor and user communities. Mike is Editor-in-Chief of the Group's *Network Monitor* research report, and serves as senior editor to two other PSOCG reports, the *Office Computing Report* and *Unix in the Office.*

INTEROP 90 covers Distributed Computing in great detail:

- ***Dr. Ralph Droms will teach a 2-day course called "Distributed File Systems and NFS" on Monday/Tuesday.***
- ***Mike Millikin will chair a session entitled "Enabling the Distributed Computing Environment" The session, S31, is on Friday at 8:30am.***
- ***Both plenary addresses on Wednesday and Thursday mornings; "Distributed Computing for the '90s—Which way to the Promised Land?" will consider the future of distributed computing. (Both start at 9:00am).***
- ***Dr. Bruce Nelson will chair two sessions; S3 on Distributed File Systems at 10:30am on Wednesday, and S13 on Remote Procedure Calls at 3:30pm the same day.***
- ***Several vendors will be demonstrating ONC/NFS technology on the exhibit floor.***

Consult your program guide for details!

ONC™/NFS®
SHOWCASE

Distributed Computing Applications Today!

Fiber Distributed Data Interface (FDDI)—A Tutorial

by Mark S. Wolter, National Semiconductor Corporation

Introduction

Some of the basic characteristics of the FDDI specification include fiber optic transmission media, ring topology, token access protocol and an architecture with a distributed management approach. The network parameters achieved by these characteristics include a 100Mbps data rate, 2–3 km distance between connections, up to 1000 connections on a network and a total distance of 200 km. The specification for FDDI is compliant with the Open Systems Interconnection.

Demand for FDDI

Mainframe computers were originally developed to support multiple users. As computers became more plentiful, the need arose to transfer data between computer systems, and proprietary computer networks were developed. The advent of mini, micro and personal computers led the need to share expensive resources between several autonomous systems, each having their own operating environment. From this need, standardized *Local Area Networks* (LANs) evolved that supported shared printers, modems, and disk drives. With the availability of this standard network, new applications not suited to the available bandwidth of existing technology LANs were developed. The need for a high speed network, FDDI, was established. Shared data intensive resources such as file systems are now used as an integral part of a node's operating system and require fast access across a network. Plotters and graphic printers, and fast interactive support of high resolution graphic workstations require the solution offered by FDDI. Different networks, each with their own characteristics, were attached to each other, creating a nightmare for a network administrator responsible for network loading and fault isolation. Better management and greater bandwidth are needed for interconnection of existing departmental LANs onto a high speed FDDI backbone.

Fiber Optic media

One of the characteristics of FDDI is the use of fiber optic cable as a communication media which has several advantages over alternative copper media. The bandwidth of coaxial cable or twisted pair wires limits the transmission speed and the distance between active connections as compared to fiber optic cables.

Fiber optic cable also has several security advantages over most alternatives. Fiber optic cable is immune to *Electromagnetic Interference* (EMI) so that it cannot be jammed by high power emissions such as radio transmitters. Fiber optics do not emit *Electromagnetic Radiation* (EMR), avoiding the capability of eavesdropping. Splicing into a fiber optic cable for the purpose of eavesdropping is extremely difficult and can easily be detected at the receiving station as an increase in signal losses at the fiber optic receiver.

As technology improves, the cost of converting from one generation standard to the next can be expensive. Installation of a new transmission medium can be particularly expensive in the conversion costs. An investment into fiber optic cable can be used for FDDI now, as well as future networks using a gigabit per second technology.

Ring topology

The topology of an FDDI network incorporates several features that allow for greater reliability and fault tolerance of a damaged network. The *dual counter-rotating ring* allows the network to be reconfigured around hardware failures that occur. Any single failure that occurs within a dual ring connection can be isolated from the ring and the integrity of the rest of the ring is still maintained.Multiple hardware failures on the dual ring will result in dividing the ring into separate rings that still function but are isolated from each other.

FDDI provides facilities which allow an FDDI network to be configured into a star topology. The support of a star topology in an office environment has several advantages to the ring topology. A typical office environment for a network consists of independent connections between each office workstation and a wiring closet, constituting a star topology. A wiring closet may be connected in a hierarchical manner to additional closets. This approach allows the flexibility to isolate stations from the network during the installation and reinstallation of office equipment which occurs on a regular basis.

Other topologies

An FDDI network allows for the support of several of these features available in a star topology. A station connection to the dual ring is known as a *Dual Attach Station* (DAS) and support the flexible reconfiguration mentioned previously. In a star topology, a station may be connected to a single ring and is known as a *Single Attach Station* (SAS). This SAS connection alone would not allow for the reliability available to a DAS, but with the constraint that it is used to attach to a *Dual Attach Concentrator* (DAC) the reliability of the dual ring may be maintained. This means that faulty SAS connections may be isolated by a concentrator, causing no damage to the remainder of the ring. *Single Attach Concentrators* (SAC) may be connected to DACs and other SACs to form a tree-structured hierarchical topology. This tree structure is connected to the trunk dual ring. (Figure 1.)

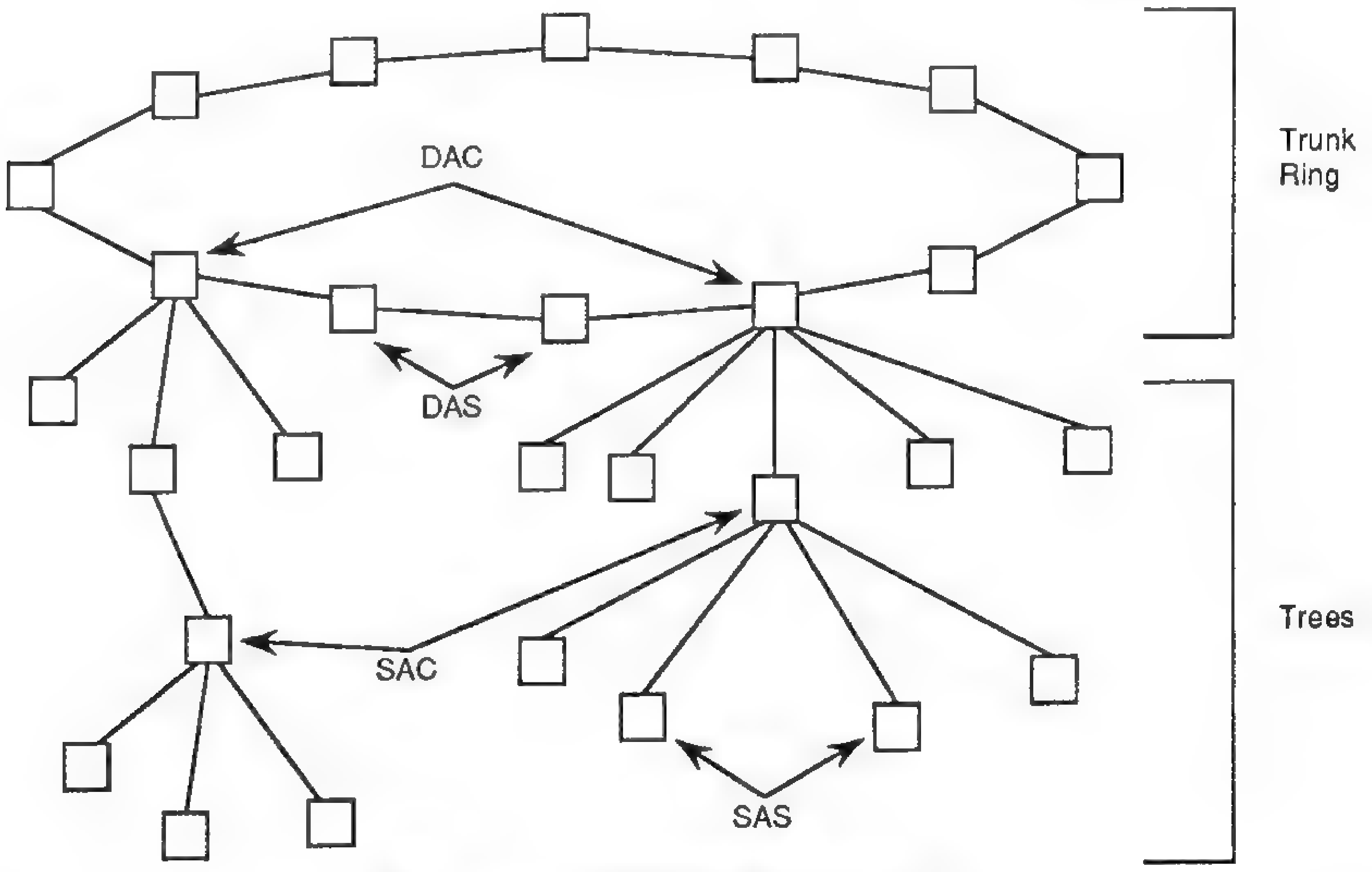


Figure 1: FDDI Topology

In the case of a catastrophic failure of a station on the trunk ring such as a power outage, the use of an optional *optical bypass switch* adds to the reliability of an FDDI network. Since this is a passive connection, it results in additional attenuation of the signal with the combined attenuation of the attached fiber segments. Care must be taken in the placement and use of this switch so as not to exceed the attenuation budget between FDDI stations.

Token Protocol

The two basic entities transmitted in an FDDI ring are called *frames* and *tokens*. Access to begin transmission on the ring is gained by capturing a special frame called a token. There is only *one* token allowed on a ring, and therefore only one station may transmit new data onto the ring at a time. A station transmits data by receiving a token, transmitting multiple frames until a timer expires, and re-transmitting the token onto the ring. While frames circulate around the ring, they are copied by the receiving station. When the frame returns to the originating station, it is stripped by that station by not being re-transmitted.

continued on next page

Fiber Distributed Data Interface *(continued)*

The circulating token protocol provides an efficient access method to the ring, particularly at high network loads. It provides deterministic access to the network, guaranteeing eventual access to the network. The immediate release of the token after transmission allows utilization of the full bandwidth of the network. The overall performance does not degrade as the number of stations added to the the network increases.

Distributed Approach

A distributed approach was used in developing the FDDI network. The primary advantage of the distribution of network functions is that the loss of any single station does not inhibit the use of the network by the remaining stations. The design incorporates distributed clocking, fault recovery, and monitoring functions to add to the reliability and management capability of the network.

The use of independent clocks on each station adds to the reliability of the network. Since each station acts as a repeater, clock signals can be recovered from the upstream neighbor's signal transmission and re-transmitted by the station's local clock. Each station requires its own clock recovery circuit, but since there is local buffering and retransmission there is no further need for the accommodation of accumulated jitter. If a clock were to fail at a given station, only that station would be affected and would be isolated by the neighboring stations from the ring.

The initialization process of an FDDI ring is done as a shared responsibility in which all stations take part. Prior to a ring becoming operational, all of the active stations participate in the parameter and token initialization. Timer expiration parameters must be set so that each station will be able to identify the failure of the ring due to the loss of a token. While the timer parameters are being set, the stations are also determining who will transmit the initial token frame. Since this approach allows any selected active station to recognize ring failure and to transmit the first token upon initialization or recovery, no single faulty station can cause the failure of the entire network.

A distributed approach is also used in the development of network management support. At the network interface level, all stations are peers and possess the same capabilities. Each station can transmit information about itself and its neighbors to a requesting station or as a broadcast message to all stations on the network. Each station can also request information and network reconfiguration. This flexibility allows for extensive diagnosis, isolation and reporting of problems on the network by any designated node at a given installation. The problems that can be identified include both physical failures and loading inefficiencies.

OSI compliance

The FDDI standard is compliant with the lower layer of the OSI protocol stack. FDDI provides services specified by the Data Link and Physical layers. The FDDI specification includes the sublayers of the Physical layer called the *Physical* (PHY) and *Physical Media Dependent* (PMD) sublayers in FDDI terminology. The Data Link layer is subdivided into the *Link Layer Control* (LLC) and the *Media Access Control* (MAC) sublayers, of which FDDI specifies the MAC sublayer. The LLC sublayer is specified as an IEEE 802.2 standard, which interfaces directly to the MAC sublayer of FDDI as well as the other IEEE 802.x MAC sublayers of other network specifications.

PMD Layer

The PMD layer includes the specification for all of the transmission media hardware. The station-to-station attenuation budget of fiber optic cables and connections are specified, as are the fiber optic transceivers. The optional optical bypass switch is also included. The fiber optic transmitter transforms an electrical signal into a signal suitable for driving a *Light Emitting Diode* (LED). The fiber optic receiver amplifies and filters the electrical output signal of a PIN (*p*-intrinsic-*n*) photodiode receiving light impulses.

MIC

The cable connector receptacle defined by the FDDI standard is called a *Media Interface Connector* (MIC). The specification provides for the alignment of optical fibers. The fiber optic plug is not directly specified, but is an implied specification as a mate to the MIC. The receptacle can be mounted on a printed circuit board. The plug and receptacle are keyed to insure the interconnectability between conforming FDDI stations. MIC A (transmit on primary, receive on secondary) and MIC B (transmit on secondary, receive on primary) provide for attachment of DAS's and DAC's to the primary and secondary fibers of the dual ring. MIC M (*Master connection*) is used in a concentrator to provide an attachment for the MIC S (*Slave connection*) of a SAS. Optical losses are supplied by the fiber and receptacle vendors and must be considered in the installation plans.

The cable plant interface specifications include fiber types and attenuation. A 62.5μm core diameter fiber with 125μm cladding is the most commonly referenced fiber size, although 50μm/125μm and 100μm/140μm sizes are also referenced. The attenuation specification allows a –11.0dB loss between stations, including cable, splices, connectors, and fiber optic switches. Typically a 1300nm wavelength multimode cable is specified as less than 2.5dB/km attenuation. The maximum cable length is defined as the maximum distance possible without violation of the –11.0dB attenuation budget which includes connector loss. For 1300nm cable, this distance is around 2 km.

The PMD includes the specification for an optional optical bypass switch, which includes attenuation, interchannel isolation, switching time and media interruption time. When the station is powered off, the switch will go into bypass mode. Care must be taken to insure that the attenuation budget is not exceeded by too many optical switches in bypass mode in series during a likely power down scenario.

PHY Layer

The *Physical* (PHY) layer handles all of the symbol based functions. A *symbol* is the basic sequence of bits which represents data and control information. The PHY layer encodes and decodes the data and control symbols. It provides serial-to-parallel and parallel-to-serial conversions. It recovers the clock signal from the upstream neighbor, and includes an elasticity buffer for synchronizing to the local station clock.

NRZI coding

The PHY layer provides the 4B/5B symbol coding function and the NRZI bit coding function. The 4B/5B (four bit to five bit) coding maps data in four bit symbols to a corresponding five bit symbol that guarantees a logic 1 bit at least every five data bits in sequence to be transmitted. Since this five bit symbol is then transmitted using NRZI (non-return to zero, invert on ones) code, a signal transition occurs every time there is a logic one and is guaranteed at least every five data bits. This guaranteed transition is required by the clock recovery circuitry, a *Phase Locked Loop* (PLL), of the downstream neighbor to remain locked to the transmission frequency and recover the clock signal. The signal transitions are also required to maintain DC balance on the receive circuitry to minimize data induced noise.

continued on next page

Fiber Distributed Data Interface *(continued)*

The 4B/5B symbol coding and NRZI bit coding provide an efficient clock encoding scheme. The 4B/5B code uses one extra bit for every four bits of data to be transmitted, or 20% bandwidth overhead for clock encoding. This is why FDDI can transmit at 100Mbps with a 125MHz clock frequency. A less efficient alternative encoding scheme might use Manchester encoding which guarantees at least one signal transition for every data bit to be transmitted, or 50% bandwidth overhead for clock encoding. The use of NRZI coding makes the highest frequency pattern or required bandwidth equal half the encoded frequency, or 62.5MHz for FDDI.

Since only half of the possible 5B codes are mapped to corresponding 4B data patterns, several of the non-data codes are used for control symbols, under the constraint that they are also combinations that guarantees a logic 1 every five bits, with the exception of the Quiet symbol. These additional symbols are divided into four groups, being line state symbols, starting/ending delimiter symbols, control indicator symbols and violation symbols. Line state symbols include *Quiet* (Q), *Halt* (H) and *Idle* (I) symbols. Q indicates the absence of any transitions and loss of clock recovery ability. H indicates a forced logical break in activity while maintaining DC balance and clock recovery. I indicates normal condition between frame and token transmissions while providing the best case conditions for clock recovery. The starting delimiter consists of a unique *J* and *K* symbol pair and is used to designate the beginning of a frame. A pair of ending delimiters *Terminate* (T) symbols is used to terminate a token frame, while a single T followed by control indicator symbols terminates all other frames. Control indicator symbols consist of *Reset* (R) or *Set* (S) symbols and are defined by the MAC layer as well as implementation dependent extensions. Finally, all remaining symbol codes are designated as *Violation* (V) symbols, some of which may be recognized as an off-alignment H symbol.

The PHY layer also has to provide an elasticity buffer to allow for variation in clock frequency from one station to its upstream neighbor. If the upstream neighbor's clock frequency is higher, I's between frames will be dropped when repeating, and if the frequency is lower, I's will be added to the transmission stream.

Line states

The PHY layer generates control symbols and detects sequences of control and data symbols by entering into a designated line state. *Quiet Line State* (QLS) is entered after a stream of 16 or 17 consecutive Q's. *Master Line State* (MLS) is entered after a stream of 8 consecutive HQ symbol pairs. *Halt Line State* (HLS) is entered after a stream of 16 or 17 consecutive H's. *Idle Line State* (ILS) is entered after as many I's. *Active Line State* (ALS) is entered after a JK symbol pair frame starting delimiter. *Noise Line State* (NLS) is entered after 16 or 17 potential noise events have occurred without entering into another line state. Normal operating conditions of a network would toggle between ALS and ILS because frames and tokens will cause entry into ALS and the I's surrounding them will cause entry into ILS. Other lines states are forced during the initialization process, the reconfiguration process or to signify a fault condition during the operation process.

In order to prevent propagation of bit stream errors and to simplify fault isolation, each PHY must provide a *repeat filter* that will not repeat V's or other NLS conditions to its transmitter.

Although this would not be required in the PHY when a MAC is included in the repeat path because the MAC performs this function automatically, some configurations of a DAS or Concentrator provide paths that do not include a MAC.

MAC Layer

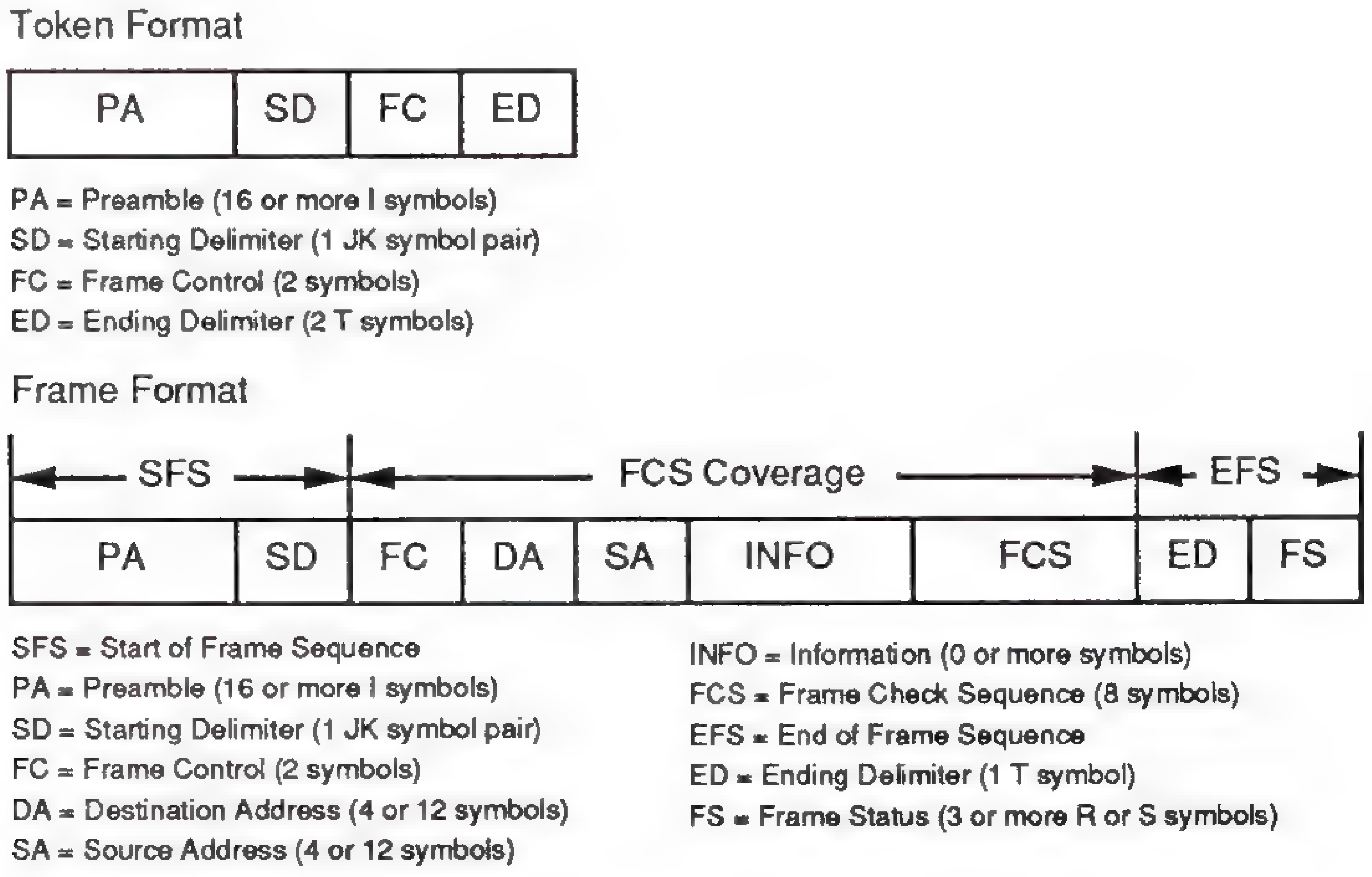
The *Media Access Control* (MAC) layer handles all of the frame based functions and controls access to the transmission media. It handles the framing of data including the starting delimiter, frame class and ending delimiter. The MAC layer generates and recognizes source and destination addresses. It also generates and verifies a CRC frame check sequence.

Tokens

The token is a unique "frame" that allows a given node to gain access to the ring for transmission of data. It consists of a preamble of 16 or more I's, a starting delimiter of a JK, a data field called the *frame control field* consisting of 2 symbols denoting that this frame is a token of a given class, and an ending delimiter of 2 T's. Two classes of tokens, *restricted* and *nonrestricted,* are defined. A nonrestricted token is used for normal operation, and a restricted token restricts all of the asynchronous (unreserved and unused) bandwidth of the network to be used by a specified set of two nodes for the remainder of their reserved connection. The MAC layer is responsible for capturing and re-transmitting the token. (Figure 2.)

Frames

Frames consist of the same preamble and starting delimiter sequence, a sequence of MAC control and data information, and an end of frame delimiter that consists of a single T symbol and a series of R and S symbols noting the status of the E, A, and C control indicator symbols as well as other implementation defined R/S indicators. The setting and reseting of the EAC indicators is administered by the MAC layer and designate whether a MAC layer condition has occurred indicating *frame check Error detected* (E), *destination Address recognized* (A) and *frame Copied* (C). The data sequence includes the frame control field, the destination and source address fields, the information field containing 0 or more symbols, and 8 symbols of the frame check sequence field. (Figure 2.)



PA = Preamble (16 or more I symbols)
SD = Starting Delimiter (1 JK symbol pair)
FC = Frame Control (2 symbols)
ED = Ending Delimiter (2 T symbols)

SFS = Start of Frame Sequence
PA = Preamble (16 or more I symbols)
SD = Starting Delimiter (1 JK symbol pair)
FC = Frame Control (2 symbols)
DA = Destination Address (4 or 12 symbols)
SA = Source Address (4 or 12 symbols)
INFO = Information (0 or more symbols)
FCS = Frame Check Sequence (8 symbols)
EFS = End of Frame Sequence
ED = Ending Delimiter (1 T symbol)
FS = Frame Status (3 or more R or S symbols)

Figure 2: Token and Frame formats

The frame control field denotes the type of frame. A void frame is a frame whose content is ignored while it is still stripped by the transmitting station, its use is explained later. MAC frames, SMT frames and LLC frames are used to transmit information between matching layer entities between stations. A reserved frame exists for use for the dependent implementation of a network.

continued on next page

Fiber Distributed Data Interface *(continued)*

MAC frames are used to initialize several MAC parameters including those explained below. Special MAC frames include the *Beacon Frame* used to localize the fault of a serious ring failure and the *Claim Frame* used to determine which station will transmit the first token and initialize the ring.

Station Management (SMT) functions implement network management facilities and use their own class of frames. A special SMT frame exists that addresses the next station, and is used by SMT for ring mapping and recovery functions. LLC frames are used for the transmission of data between the upper layers of the OSI stack and serve as the main data stream for network connectivity. LLC frames can be sent as synchronous or asynchronous frames. Synchronous frames are frames that are sent as part of a station's predetermined guaranteed bandwidth and are top priority frames. Asynchronous frames use the unreserved or unused bandwidth of the network and allocation of this bandwidth is provided to the stations in a round robin fashion.

Addressing

The destination and source address fields can be either 16 or 48 bit addresses. All stations are required to fully support the 16-bit address frames, repeat all 48-bit address frames and support the 48-bit Claim, Beacon and Broadcast address frames. Broadcast addresses consist of all ones. Null addresses of all zeros are not recognized by any stations as its address. The first bit of the destination address field denotes whether a frame is an individual or group address, while the second bit of the 48 bit address denotes whether the address is a universally or locally administered address. The first bit of the source address always denotes an individual address.

The information field consists of 0 or more eight bit octets of data transmitted as symbol pairs. The frame check sequence field consists of a 32 bit CRC polynomial commonly used for communication protocols that encapsulates the entire data sequence, including the frame control, addressing and information fields, and the frame check sequence field itself.

Timers

Three timers and a counter are used by the MAC layer to regulate the operation of the ring. The *Token Holding Timer* (THT) controls how long the station may transmit asynchronous frames. The *Valid-Transmission Timer* (TVX) is used to recover from transient ring error situations. It times the delay since the last valid transmission, and a void frame may be used to reset the TVX in the absence of a normal data frame. The *Token Rotation Timer* (TRT) controls ring scheduling during normal operation and to detect and recover from serious ring error situations. The *Late Counter* (*Late_Ct*) counts the number of TRT expirations since the MAC was reset or a token was received.

Three additional counters are used to aid in problem determination and fault location. The *Frame_Ct* is a count of all complete (non-fragment) frames received. The *Error_Ct* is a count of frames received with a frame check error and without the E indicator already set. The *Lost_Ct* is a count of all instances that while receiving a frame or token an error occurred that threatened the integrity of the frame itself, such as the reception of a V symbol.

Claim process

A station with data to transmit gains access to the ring by capturing a token. It continues to transmit multiple frames until it is finished or its available time expires, and then re-transmits the token onto the ring. While frames circulate around the ring, they are recognized by the receiving station which sets the A indicator.

If the station can, frames are copied to that station's available buffer space after which the station also sets the C indicator of the frame during re-transmission. When the frame returns to the originating station, it is stripped by that station by transmitting I's immediately following its recognition, i.e., the source address field.

During the Claim process, all stations in a ring agree upon a common *Target Token Rotation Time* (TTRT), the station with the lowest bid TTRT winning. Some stations are also assigned a synchronous allocation time that guarantees that station an allotted transmission bandwidth. A station's synchronous allocation time is based on that ring's implementation and set by an SMT-to-SMT communication. Each time a token is received, *Late_Ct* is reset to zero and TRT is reset to TTRT and begins to count down. If another token is received before TRT expires, both asynchronous and synchronous transmission may take place. At this time, THT is set to TRT and TRT is reset to TTRT to begin timing the next rotation time. THT is then the unreserved and unused bandwidth that is available for asynchronous transmission, and asynchronous frames may be transmitted until THT expires. An asynchronous priority scheme also exists that requires THT to be of greater value than a threshold value *T_Pri(n)* before a frame of priority level (n) may be transmitted.

If instead TRT expires before the arrival of another token, *Late_Ct* is incremented to one and only Synchronous transmission can take place for its allocated time upon reception of the token. If TRT expires a second time before receiving a token, *Late_Ct* is incremented to two, the token is considered lost, and a new Claim process will begin. This protocol guarantees an average TRT (or average synchronous response time) not greater than TTRT, and a maximum TRT (or maximum synchronous response time) not greater than twice TTRT. This protocol also guarantees that the asynchronous bandwidth is not taken entirely by one station, but that every other station has a chance to use this bandwidth in a round-robin fashion before a given station gets another chance.

SMT Specification

The FDDI standard also includes a *Station Management* (SMT) specification which allows for the necessary network management functions for performance monitoring, fault detection and error recovery. SMT is subdivided into three areas: *Frame-Based Management, Connection Management* (CMT) and *Ring Management* (RMT). SMT provides frame-based functions that gather information about and exercise control over the FDDI network. CMT manages the PHY components and their interconnections, and uses MAC and PHY entities within a station to achieve logical attachment of that station to the ring. MAC layer components and the rings to which they are logically attached are managed by RMT. (Figure 3.)

Management frames

SMT frame-based management services include the use of several types of SMT frames for performing various functions. *Neighborhood Notification* (NN) uses *Neighborhood Information Frames* (NIFs) to determine its *Upstream* and *Downstream Neighbor Addresses* (UNA and DNA), to provide supplemental duplicate address detection, and to verify the operation of local MAC receive and transmit paths in the absence of any other traffic. NIFs provide information for resolving network faults and constructing logical ring maps. *Status Report Frames* (SRFs) provide optional information by reporting network parameter conditions and connection events.

continued on next page

Fiber Distributed Data Interface *(continued)*

Optional *Parameter Management Frames* (PMFs) are used by a protocol to operate on all SMT *Management Information Base* (MIB) attributes to allow remote management of station attributes, such as the synchronous bandwidth allocation of each station. *Station Information Frames* (SIFs) provide station status obtained remotely by polling stations for connection and configuration parameters, and statistical operation information. *Echo Frames* provide for SMT-to-SMT loopback testing on a ring. *Root Concentrator Polling* allows a node in a concentrator tree to determine where it is within a tree of concentrators and where it resides in terms of global connectivity by using any of the SMT request/response protocols. *Extended Service Frames* (ESFs) are defined for extending new SMT frame based protocols and require a unique identification parameter. *Request Denied Frames* (RDFs) are defined to allow backward compatibility by allowing new protocols to easily identify incompatible stations in a mixed environment.

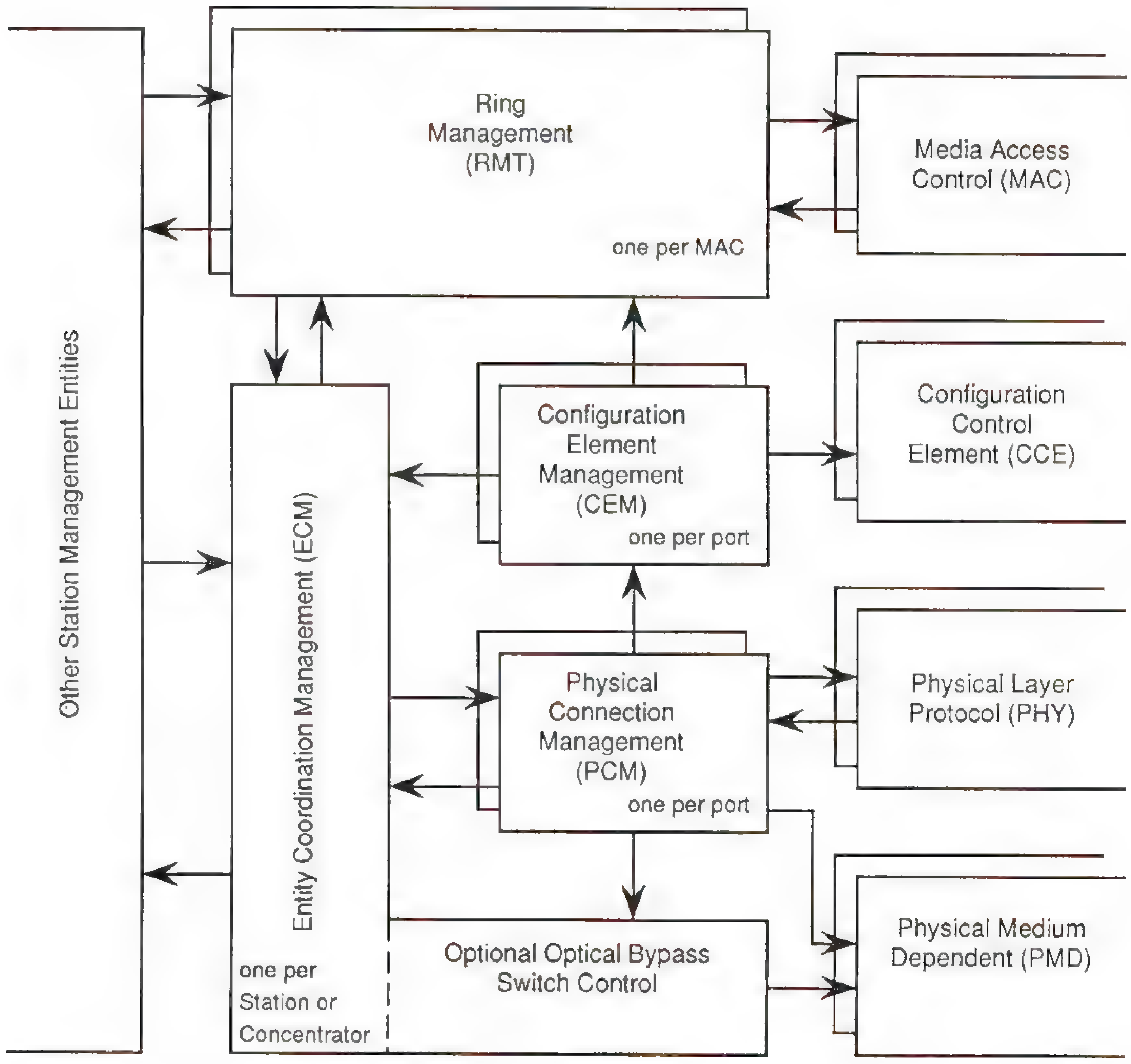

Other Station Management Entities
Ring Management (RMT)
one per MAC
Media Access Control (MAC)
Entity Coordination Management (ECM)
one per Station or Concentrator
Configuration Element Management (CEM)
one per port
Configuration Control Element (CCE)
Physical Connection Management (PCM)
one per port
Physical Layer Protocol (PHY)
Optional Optical Bypass Switch Control
Physical Medium Dependent (PMD)

Figure 3: Station Management interfaces

Connection Management

Connection Management (CMT) operates the insertion and removal of PHY and PMD entities called *Ports* to the ring, and the connection of ports to the MAC entities. CMT is further subdivided into three areas, including ECM, PCM, and CFM. *Entity Coordination Management* (ECM) is responsible for the media interface to the FDDI network, including the coordination of the activity of all the Ports and the optional optical bypass switch associated with that station or concentrator. After completing optical bypass switching, ECM signals *Physical Connection Management* (PCM) when the media is ready to begin initialization of the PHY. ECM also coordinates the *Trace* and *Path Test* functions, which are used to localize a *Stuck Beacon* condition and test the path to the nearest upstream MAC, respectively.

PCM initializes the physical connection between the Port being managed and another Port, such as in an adjacent station or concentrator on the FDDI ring, by signaling between these Ports upon a request from the ECM. Signaling is done by transmitting a continuous stream of symbols until the neighbor responds, which transitions to a new line state and the station begins to transmit a different stream of symbols awaiting the next response. PCM sequences through a number of bit signals to communicate such connection information as the port type (A,B,M,S), the compatibility of this connection, the duration of a *Link Confidence Test* (LCT), the availability of a MAC for LET, the failure of a LET, the assignment of a MAC for Local Loop testing, and the assignment of a MAC to this port for ring operations as a tree or peer MAC. Once the connection has been verified completely by this procedure, a response is issued to *Configuration Management* (CFM). Through the use of this signaling feature, CMT on one station can force its neighboring CMT to a known state by transmitting a series of symbols, aiding in the isolation of faults in a ring.

Configuration Management

Each Port in a station or concentrator has a PCM entity associated with it. Each PCM has a *Configuration Element Management* (CEM) entity associated with it. Each CEM controls an associated *Configuration Control Element* (CCE), sometimes referred to as the configuration switch, which physically connects the MAC and Port entities. *Configuration Management* (CFM) collectively refers to all of the CEMs in a station or concentrator and manages the configuration of the MAC and Port entities therein. CFM supports the configuration of all types of stations and concentrators including DAS, SAS, DAC, and SAC through the use of four different CEMs supporting A and B connections to a dual attach ring and S and M connections for all single attachments. There is also a single CEM specifically used for coordinating the connection of all of the MACs within a node called *MAC Placement Management.* Each CCE is required to provide a primary path connection to its associated Port, and may optionally include the capability for connecting the secondary path. The allowance of a local path that may be used for connecting a dedicated MAC to the Port also exists. CFM is also responsible for ensuring that a reconfiguration of the ring does not create any undesirable frames on the ring. This includes *ring scrubbing,* or the removal of frames from MACs that are no longer part of the token path, non-concatenation of frames, or sourcing at least 16 I's before connecting a new path into the ring, and the avoidance of concurrently processing A and B Port CEMs which may result in the entrance into undesirable states.

Ring Management

Ring Management (RMT) is responsible for receiving status from the MAC and CMT and reporting the status of the MAC to SMT. RMT identifies a Stuck Beacon by the expiration of a timer, and then *Directed Beacons* are sent to see if the problem still remains unresolved. If it does, RMT initiates the Trace function which uses PHY signaling to identify the fault domain, which results in a Path Test in all nodes in the fault domain. A number of conditions can be used by RMT to detect duplicate addresses during the Claim and Beacon processes which could result in the processes never being resolved. Resolution of duplicate address problems that prevent *Ring_Op* is done so that MACs with duplicate addresses will not prevent communication by the entire FDDI ring. This can be done by changing the MAC address, configuring the MAC to lose the Claim process and disabling its LLC services, or removing the MAC with the duplicate address from the ring.

continued on next page

Fiber Distributed Data Interface *(continued)*

Summary of benefits

FDDI is a fast reliable network protocol which provides many unique network management functions. Its fiber optic transmission media allows longer, more secure connections between stations. Its efficient use of the available bandwidth insures maximum throughput. Its token method provides deterministic access services which will not degrade as the number of connections increase. It provides advanced capabilities with provisions made for extension to the protocol.

References

[1] American National Standards Institute (ANSI), "FDDI Token Ring Media Access Control (MAC)," American National Standard ASC X3T9.5, 1986.

[2] ANSI, "FDDI Physical Layer Protocol (PHY)," Draft Proposed American National Standard, ASC X3T9.5, Rev. 13,1986.

[3] ANSI, "FDDI Physical Layer Medium Dependent (PMD)," Draft Proposed American National Standard, ASC X3T9.5, Rev. 7.3, 1988.

[4] ANSI, "FDDI Station Management (SMT)," Working Draft Proposed American National Standard, ASC X3T9.5, Rev. 6.1, 1990.

INTEROP.
FDDI
1990

MARK WOLTER has worked at National Semiconductor for over six years, and is an application manager for FDDI products. Prior to his work with FDDI, he worked on the design and implementation of disk data and networking controller chips. He played a leading role in the development of a design methodology incorporating the use of CAE tools and standard cells in the development, simulation, verification, and system modelling of complex VLSI chips. He holds a B.S in Electrical Engineering, with a concentration in Computer Engineering, from Cornell University.

At INTEROP 90, attend S20: "FDDI Design Issues" on Thursday October 11, at 10:30am. Also, don't miss the FDDI demonstrations in the exhibition hall.

Coming in Future *ConneXions:*

"Profile: NORDUnet" by Mats Brunell, SICS.

"Overview of Path MTU Discovery" by Jeff Mogul, DEC WRL.

"Components of OSI: X.25" by Derek Vair, Software Group.

"Components of OSI: ASN.1" by Derek Robinson, COS.

"Congestion Avoidance" by Paul McKenney, SRI International.

"Network Management Directions"
by Karl Auerbach, Epilogue Technology & Denis Yaro, Sun.

"FTAM, FTP and NFS in an Enterprise Network"
by Eric Fleischman, Boeing Computer Services.

"The Border Gateway Protocol" by Yakov Rekhter, IBM.

"Policy Routing" by Martha Steenstrup, BBN.

"Inter-Domain Routing" by Robert Woodburn, Sparta, Inc.

"Press Here for the Internet" by Daniel Dern.

"The Intermail Service and The Commerical Mail Relay Project"
by A. Westin, A. DeSchon, J. Postel & C. Ward, USC-ISI.

"IP Multicasting" by Steve Deering, Xerox.

...plus book reviews, conference announements and much more.

CONNEXIONS™
The Interoperability Report

October 1990 | Special Issue INTEROP 90 Backgrounder | Volume 4 No. 10

*ConneXions—
The Interoperability Report tracks current and emerging standards and technologies within the computer and communications industry.*

In this issue:

Distributed Computing Environments 2
FDDI 16
ISDN: Why use it? 27
SMDS 34
X Windows programming 38
Wireless Networking 42
Gigabit Networking 46
OSI/NM Forum 48
DECnet/OSI Phase V 56
For further reading 63

ConneXions is published monthly by Interop, Inc., 480 San Antonio Road, Suite 100, Mountain View, CA 94040, USA. 415-941-3399. Fax: 415-949-1779.

Copyright © 1990 by Interop, Inc. Quotation with attribution encouraged.

ConneXions—The Interoperability Report and the *ConneXions* masthead are trademarks of Interop, Inc.

ISSN 0894-5926

From the Editor

Welcome to INTEROP 90. This special edition of *ConneXions*—over twice the size of a "normal" issue—is designed to complement the conference program with "background reading" on a few select topics. Needless to say, with 46 sessions and around 200 speakers, an issue containing articles on *every* topic would turn into a book. We will of course continue to cover conference topics in future issues, so I encourage you to sign up for a subscription now and ensure that you stay informed in this rapidly changing field.

The first 6 articles in this issue address emerging technologies which are being showcased at INTEROP 90, *Distributed Computing Environments, FDDI, ISDN, SMDS* and *The X Window Sytem*. The articles are by Mike Milikin, Mark Wolter, Dory Leifer, Larry Hughes/Steve Stamper and Wayne Dyksen/Tim Korb respectively. You'll see and hear a lot about these topics this week—in tutorials, conference sessions, informal BOF sessions, and through numerous special technology demonstrations on the exhibit floor. Check your conference program for further details.

Networks are currently being built using several forms of electromagnetic radiation in place of conventional wire. Richard Allen gives an overview of these technologies in an article on page 40.

Craig Partridge describes the state of very high speed (gigabit) networking in a brief article starting on page 44.

Network management is a major topic at INTEROP 90. Reflecting this, Bruce Murrill presents a profile of the OSI/NM Forum. We will follow this next month with an article entitled "Network Management Directions" by Karl Auerbach and Denis Yaro.

"Multiprotocol Internetting" is the overall theme for INTEROP 90. We envision the future of networking as not necessarily being dominated by one protocol suite (such as OSI). Instead, a number of diverse technologies will need to coexist and interoperate. This is a major technical challenge, and it is addressed in several sessions. One existing architecture which will have to be part of this protocol spectrum is DECnet, or *DECnet/OSI Phase V*. Carl Malamud examines the future of DECnet and its relationship to OSI.

At the end of this issue you'll find a list of suggested articles for further reading.

Over the next several months, we will follow up with articles from other INTEROP sessions and BOFS. A list of some of these articles can be found on page 26. We hope you enjoy your stay in San Jose!

Subscribe today!

ISDN: Why use it?

by Dory Leifer, University of Michigan

Introduction

There seem to be two diametrically opposed opinions about the role that *Integrated Services Data Network* (ISDN) should play in building modern enterprise networks. At one extreme, some *Bell Operating Companies* (BOCs) are marketing ISDN as a single cure-all, a network which will finally obviate the need for the diverse collection of often incompatible local area networks, routers, and bridges along with the expensive installation of cables such as coax and fiber which connect these devices. According to this "party line," most of the data community would be much better off if they exchanged data traffic through the phones on their desks and accepted unshielded twisted pair as the ultimate standard for the subscriber-network interface. This attitude is generally not accepted very well within the data networking community; after all, what do Bell marketing people know about the needs of data users to make such bold statements?

Unfortunately, many of us from the data networking community who criticize people who tout ISDN as the ultimate network often take another extreme approach that ISDN is good for very little. This is an unfortunate circumstance, because many of the critics belong to a group that has the talent to turn ISDN into something very useful; very useful, not in the sense that it will be a replacement for all existing network technologies, but that it will serve as an integral piece in the enterprise networking puzzle.

In many ways, ISDN is able to provide capabilities that none of our current data solutions can, not because of the bandwidth it can deliver when compared to a typical multi-megabit LAN, but because ISDN has the potential to deliver connectivity to users who could never, in the past, be economically connected to a data network. For about double the price of a voice line, a user has access to a network that provides economical bandwidth and is maintained by redundant switching equipment and a crew of people willing to do repairs at all hours if something breaks.

ISDN, although a clear descendant of the current voice network, has capabilities that allow it to operate in applications far more sophisticated than what the current voice network can support. For example, the concept of an out-of-band signaling channel between the user and network is distinct to ISDN. Out-of-band signaling allows control information, user messages, and even network management information to be passed between users and the network. ISDN is much more than a replacement for a dial-up modem; it is a network designed to transport data and provide user access to signaling capabilities of the public switched network.

Solutions in search of a problem

Probably the single most effective way to exploit the capabilities of ISDN is to facilitate *interworking* between the ISDN and existing networks. This seems almost too obvious to even mention; after all, when FDDI was introduced, how many developers tried to create an entirely new network architecture and applications based on it? Yet, many people marketing ISDN products are looking for specific applications based directly on ISDN. These applications are often "a solution in search of a problem." There have been some attempts to provide interoperability between ISDN and existing networks, but most have centered around terminal service and not peer to peer networking. ISDN-compatible networking products are evolving, but very few products are actually available today.

continued on next page

ISDN: Why use it? *(continued)*

The University Campus environment

The University of Michigan faces many networking challenges associated with a large university. The need for connectivity is growing and, at the same time, users are becoming geographically dispersed. There are three salient forces perpetuating this. First, physical space in the campus core is a dwindling resource as the campus-based community grows. This population growth forces new departments into off-campus buildings not equipped to easily handle data communications. Second, universities are assuming a role as community network providers, attaching businesses, community colleges, research parks and residences to the campus network infrastructure. Providing service to a widely distributed user base economically can be quite challenging since often there are very few workstations at a given site. This isolation drives up the cost per attachment dramatically when traditional leased lines and routers are deployed. The residence is the most glaring example of this because it usually contains only a single workstation. A third reason is related to the costs of computing hardware. As personal workstation prices drop, users are no longer dependent upon central organization-provided computing resources.

Users are free to purchase equipment and use it where they prefer to work, often at home. What users find very quickly is that network services available to them once they leave the campus are often inadequate for many applications requiring network connectivity and buying this connectivity is out of the reach of many users who are often students; the monthly charges for traditional private line service, even at 56Kbps, could pay the rent for many users who are able to afford a personal workstation.

Catch 22

ISDN is an option to provide connectivity to campus data networks from almost anywhere within a region where the local BOC provides service. This region often includes residences and locations where small groups of users are located. ISDN deployment plans and tariffs are the probably the two most important factors in deciding whether ISDN has a place in a university community. These factors translate into "Can I get it to my house?" and "How much will it cost me?" Deployment is often a tricky issue because it always results in some type of Catch 22: users are not interested in the service unless it covers wide areas and phone companies are not interested in deploying services without a base of users. Applying ISDN as an access to a campus network does have an advantage over several other uses of ISDN in that campus access usually implies local access within a community. Most current implementations of ISDN are islands; i.e., data calls cannot extend beyond a single local central office. Many other uses of ISDN require that the ISDN be a fully-connected wide area network.

ISDN versus private lines

An additional motivation to deploy ISDN is to replace existing facilities when favorable tariffs apply. Although there are very few tariffs to date for ISDN service, we can make some general assumptions about the way the service will be tariffed. It is important to note that ISDN is a switched service which usually implies that the recurrent basic connection charges are small compared to the usage fees. For lines that are used sporadically, one can expect that ISDN will be less expensive than a private line which has no usage-associated costs but, on the other hand, has a much larger monthly charge. What often shifts the balance between switched and private line service in favor of ISDN is that many of the initial ISDN customers have Centrex contracts that do not include charges for ISDN data calls that are made within the Centrex system.

These "intercom" calls usually involve only a single central office switch although the actual distance of the call could be several miles. This type of charging scheme allows calls that are always active or "nailed up" to be made at far lower costs to the user than the equivalent capacity private line, sometimes saving as much as 80%. This situation will probably not persist for very long since nailed-up connections consume disproportionate amounts of the statistical capacity available in a switch which is normally engineered for typical voice traffic.

ISDN can also work in tandem with existing private line service by providing on-demand bandwidth to back-up a failed leased line or in parallel to provide additional bandwidth to handle spill traffic to help relieve a congested network. This availability of supplemental bandwidth can allow a network to be under-engineered since more capacity can be dynamically added in seconds to meet the demands of peak traffic. With the latest announcement of ISDN "H" (384Kbps or 1.536Mbps) channel support by some of the major interexchange carriers, supplementing or backing-up T1 networks with ISDN calls becomes viable.

IP over ISDN

Interworking ISDN with TCP/IP is essential in making a service which will support a wide variety of networking applications to the Internet community. IP over ISDN is very applicable on the university campus because Internet protocols are pervasive within the academic community; moreover, IP over ISDN is not limited to this environment. For example, in some areas of the world, leased facilities are not as easy to obtain as they are in the United States. Such is the case in many parts of Europe where users are encouraged to use public networks instead of private line service. Given the extensive plans for ISDN in the future 1992 telecommunications infrastructure, ISDN could be an important medium for carrying IP for European users of the Internet.

The Internet/ISDN Gateway Project

The Internet/ISDN gateway is a project at The University of Michigan which allows interworking of Internet protocols with ISDN. Several goals are identified:

- Provide transparent access to TCP/IP-based campus networks. The user and higher-layer network entities should not be "aware" of the ISDN.
- Provide concentration to allow many ISDN users to access a single gateway simultaneously.
- Insure user authenticity by scrutinizing the calling line identification delivered by ISDN. This would allow users to be assigned a permanent IP address which would be secure from impostors.
- Efficiently allocate resources to statistically allow many more users to use IP services than the number of actual "ports."
- Provide flexibility to use multiple channels in parallel to increase throughput when facilities are available.
- Support a variety of common workstations on the remote end.
- Support attachments of small remote IP subnets.
- Implement in a modular fashion to allow easy interfacing with off-the-shelf software and hardware.

continued on next page

ISDN: Why use it? *(continued)*

Configuration

The following diagrams illustrates the current prototype configuration at The University of Michigan. ISDN service is provided by a Michigan Bell-owned AT&T 5ESS. A 386 PC acts as a gateway between the ISDN and the Internet. This gateway is capable of receiving or initiating up to six incoming or outgoing circuit- switched connections and manages these connections dynamically. The gateway routes IP datagrams for remote machines connected to ISDN. Users use ISDN basic rate interfaces installed in their homes to connect PC-compatible machines to the network. Telephones can be attached to the PC to allow standard voice service to be used simultaneously with the data connections or, alternately, two circuit-switched 64Kbps connections can be used for data to increase throughput.

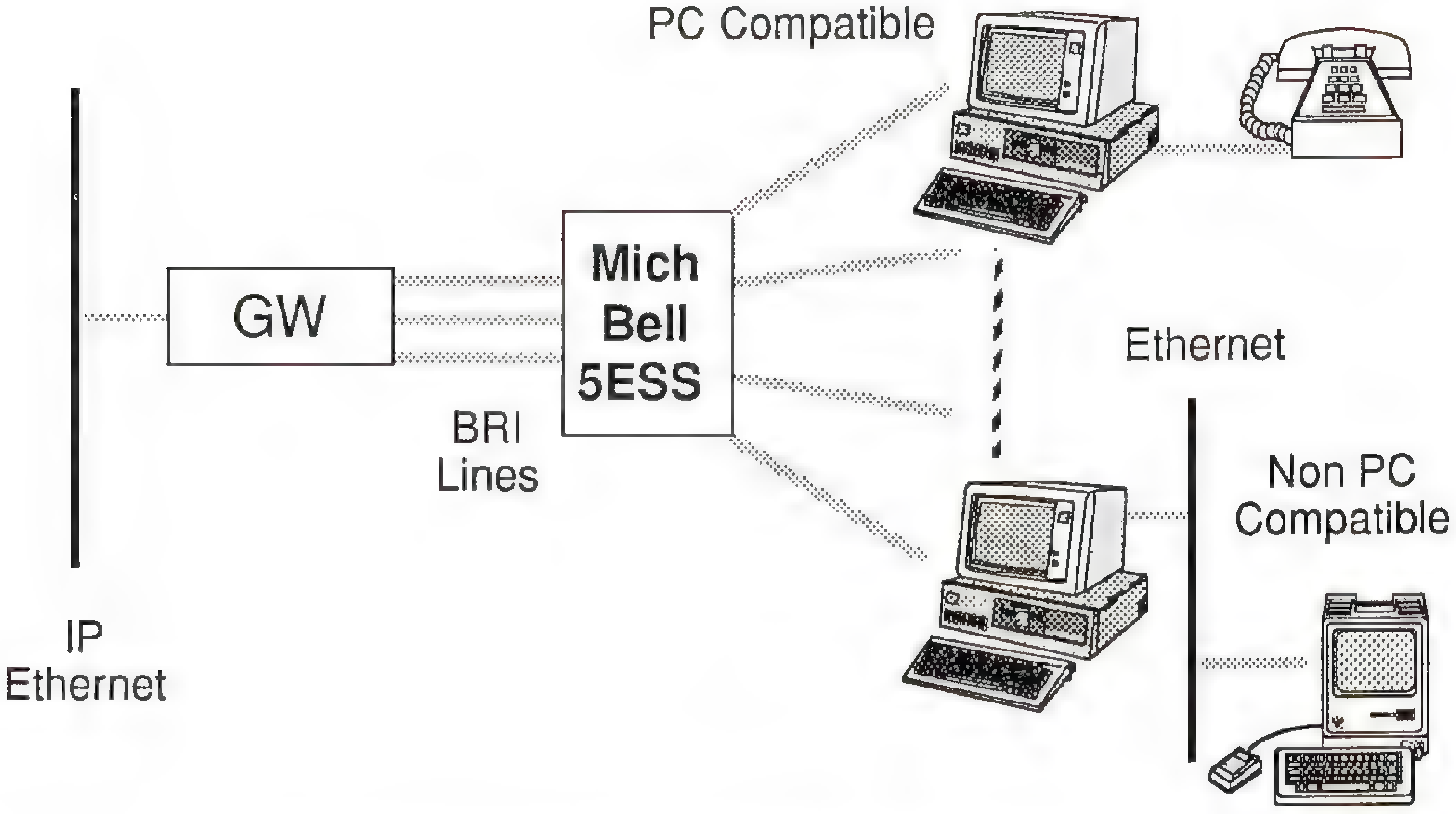


Figure 1: University of Michigan Prototype Configuration

Design

The Internet/ISDN gateway is based on an IBM PC-compatible machine running MS-DOS. This platform was chosen because it supports ISDN interface cards. The gateway is based on an FTP Software, Inc. *packet driver* which appears as either a SLIP or Ethernet interface to applications above it. The PC-based TCP/IP package KA9Q (Phil Karn, et al) is used unmodified to handle routing functions between the ISDN and Ethernet. The ISDN packet driver is actually far more complicated than a driver for a LAN card. The packet driver is responsible for access to a particular IP subnet. KA9Q "sees" this subnet as a single virtual LAN but in reality it is up to the ISDN driver to initiate and clear calls to remote stations.

The gateway and remote machines are almost identical in terms of hardware and software. On the remote machine, users are free to run applications that interface with the ISDN packet driver such as FTP Software, Inc. PC/TCP (FTP, Telnet, etc.), Graphic Software Systems (GSS) X-View (X-Windows for the PC), and KA9Q—or they can configure the remote machine to provide trivial routing functions to a real IP-Ethernet. This routing function runs in the background and frees the PC to run applications. We have successfully attached UNIX workstations, other PCs, and X-Terminals to Ethernets that are served solely by an ISDN gateway.

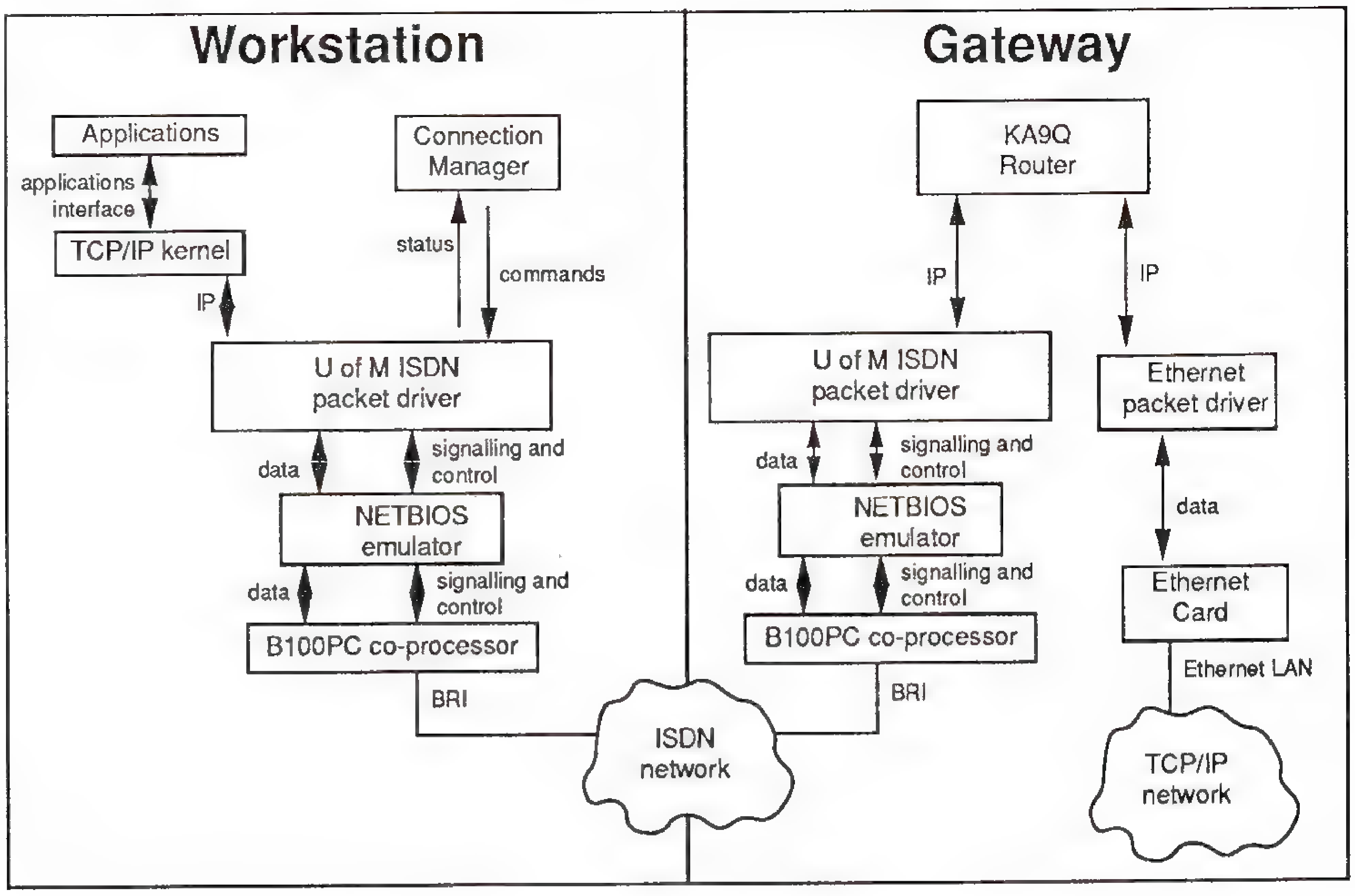


Figure 2: Configuration Details

Performance

The prototype configuration transfers files using *File Transfer Protocol* (FTP) at 6.7K bytes per second on a single 64Kbps connection. Using two simultaneous connections, throughput in excess of 11K bytes per second has been observed with FTP as well as with the *Andrew File System* (AFS). This latter speed is approximately 50 times the throughput of a 2400bps modem.

Circuit versus packet

Probably the most important consideration is whether the *Circuit-Switched* (CS) or *Packet Switched* (PS) capabilities of ISDN are to be used since ISDN provides access to both CS and X.25 PS networks. The major tradeoffs for using CS bearer services is that in exchange for a "clear channel" bit pipe which is not examined, altered or delayed by ISDN, the user must employ strategies to minimize the time that the connection is held as well as provide concentration. PS services have the advantage of a virtual circuit capability which provides concentration of many connections into a single interface. Another advantage of PS services is that there are usually very small charges for connections that are logically active but have little or no traffic traversing them. The scheme we chose for our gateway involves only CS services for several reasons:

- A 64K channel for CS services is more prevalent than it is for PS services. Access to PS bearer services is usually included on basic rate lines using only the slower 16Kbps "D" channel, while CS capability at 64Kbps is usually present.
- X.25 packet switches are often engineered to provide for terminal service, and consequently tend to support small windows and small packet sizes.
- Transmission latencies are on the order of bit times for circuit switches but can be significant across packet switched networks.
- Circuit switched networks provide guaranteed bandwidth to allow the full 64Kbps to be available for the duration of a call.
- Protocol compatibility between the user equipment and ISDN is simpler for CS than for PS because CS networks do not examine user data once a call has been established.

continued on next page

ISDN: Why use it? *(continued)*

Connection management

Connection management is an issue whether ISDN CS or PS services are used. Connection management is a scheme that controls the boundary between a network that provides connection-oriented services like ISDN circuit and packet bearer capabilities and a network that provides connectionless service like IP. This is not a new issue in the Internet community as X.25 has been used to carry IP traffic for years. RFC 877 specifies connection management algorithms for carrying IP over public X.25 networks. Further insight can be gained by reading ISO 8473 which discusses considerations for a subnetwork dependent convergence function to allow the *Connectionless-mode Network Protocol* (CLNP) to operate over X.25 Packet Level Protocol.

Connection management with CS is slightly more complicated than over X.25 since circuit switched connections consume real resources such as network capacity, user access channels, and often user dollars whenever a circuit is connected regardless of the amount of traffic flowing over the connection. This is quite different from the analogous case for a packet network for which a connection is defined as a *virtual circuit* which consumes little resources when idle; moreover, the number of virtual circuit connections a user can initiate is not directly related to the channel capacity of the subscriber-network interface.

How well a gateway handles connection management is closely related to the level of transparency between the two networks on either side of the gateway. Ideally, users of connectionless networks should not be aware that datagrams must traverse a connection-oriented network. This can be a difficult goal to achieve because of the inherent differences between connectionless and connection-oriented networks. However, heuristics can be employed to minimize the impact of these differences. Three general connection management strategies could be employed:

- *Manual:* The user must issue a command on the workstation in order to set up a call. Once a connection is established, IP datagrams can be exchanged over ISDN. When the user is finished, another command is issued to tear down the connection. This mode of operation is almost identical to using a traditional terminal and a dial-up modem.
- *Dynamic Asymmetric:* A user is connected to the network automatically when his/her workstation needs to initiate an exchange of IP datagrams.
- *Dynamic Symmetric:* A user is connected automatically upon demand but is also called by a gateway when the IP network needs to send datagrams to the user. This scheme provides maximum transparency because it allows machines attached with only an ISDN line to be accessible from the rest of the network. The cost for this transparency is the complication involved in managing connections, deciding when to initiate calls and tear them down.

The ISDN gateway software was originally designed for manual connection management but now employees dynamic symmetric. The number of users we have currently in the trial is inadequate to gauge how well the heuristics perform under heavy demand.

Addressing

In order for IP over ISDN to operate with dynamic connection management, a translation scheme must exist to map IP addresses to the E.164 ISDN numbering plan.

In some ways, this is analogous to determining the hardware address of an Ethernet interface which is associated with a particular IP address. The normal scheme of "ARPing" to get a subnetwork address is not applicable to an ISDN network because of the lack of multicast capabilities. Even if there existed multicast capabilities, it would be unreasonable to broadcast to a group of thousands. There are at least three options to handle this problem of mapping addresses:

- *Static table:* The gateway serving ISDN users has a static table which contains entries for each user's E.164 address and the corresponding IP network address and mask. When the gateway gets packets for a particular remote IP network, the gateway searches the address table and makes the appropriate ISDN call.
- *Derived:* A function is determined to map one-for-one between IP and E.164 addresses. For example, a class B subnet could contain all possible four digit extensions for a particular exchange. An advantage of this scheme is that every ISDN user has an implied IP address but a serious disadvantage is that the scheme only works for small ISDN islands since there are not enough bits in an IP address to fully encapsulate E.164.
- *Directory Query:* ISDN/IP gateways could use X.500, for example, to obtain information about a particular user and his or her associated network by specifying either the E.164 or IP network address. The advantage of this scheme is mostly operational since it allows many gateways to gain information from a single logical source—in reality this source could be distributed among several organizations to allow individual control of address domains.

Link Layer

Once a circuit switched connection is established, the ISDN transparently passes bits of user data through the network. Consequently, as in the case of a leased line, the user must choose a compatible link layer on both ends of the connection. The *Point to Point Protocol* (PPP) is usually the preferred method of passing IP on leased lines, however, ISDN user equipment is often designed to provide reliable communications as well as compatibility with terminal adapters. As a result, a standard has emerged as the link layer protocol for circuit switched ISDN connections in North America. CCITT V.120 which is based on LAPD specifies a protocol for providing flow control, multiplexing, and rate adaption (communicating between devices at different speeds). On many ISDN co-processor cards, the link layer handling is implemented directly on the card and is not under user-control so debating the issue of ISDN-IP link layers may be moot; be thankful that there is a standard.

Conclusion

There are many reasons to use ISDN data communications. ISDN may become widely used in the Internet as IP over ISDN is standardized and as Bell companies continue to deploy services. As ISDN evolves from being a series of islands into a well connected wide-area network, many more applications of IP over ISDN will emerge allowing connections to the Internet from areas that could not be served by private lines and traditional routers.

DORY LEIFER is a Systems Research Programmer for The University of Michigan Information Technology Division and the Merit Computer Network. Dory holds a B.S. in Computer Science from Rensselaer Polytechnic Institute and is currently completing an M.S. in Industrial and Operations Engineering from The University of Michigan. He is an active member of the NIST North American ISDN Users Forum and a member of the IEEE Communications Society.

Attend S15: "Putting ISDN to Good Use" on Wednesday October 10, at 3:30pm. Also, look for the ISDN demos in the exhibition hall.

Switched Multimegabit Data Service (SMDS)

by Larry Hughes, AT&T and Steve Starliper, Pacific Bell

Introduction

It's somewhat unusual in our experience to see pre-introduction interest as high as that already demonstrated in *Switched Multimegabit Data Service* (SMDS). Every Regional Holding Company is currently running or planning a *Metropolitan Area Network* (MAN) trial, and they all are working through Bellcore to set the standard for SMDS that would establish it as a national service.

SONET

SMDS is a proposed service offering public connectionless packet-switched data service over a wide area. When introduced, SMDS will provide 1.544Mb/s (DS-1) or 44.736Mb/s (DS-3) access to a fiber-optic-based switched network. Eventually, SMDS will offer *Synchronous Optical Network* (SONET) transmission rates up to 150Mb/s. The standard has also been defined to include voice and video, although SMDS will initially offer data only.

In that description lie several keys to the keen early interest in SMDS: high speeds. Fiber-optic-based transmission. Instantaneous access. Public network. They all suggest solutions to problems that have been identified over the past few years by end-users who have come to rely more and more heavily on private networks for using and sharing information within a community of interest.

The LAN marketplace

The boom in local area network (LAN) sales over the past half dozen years is clear indication that, in both the synchronous and asynchronous worlds, data users want powerful capabilities, and they want them reasonably close at hand.

Statistics on the growth of the LAN market shouldn't be interpreted too simplistically, as they tend to track a variety of supporting technologies, but the study released this year by Salomon Brothers is indicative of how steep the growth curve is expected to be. The study depicts the worldwide LAN marketplace as $4.68 billion in 1989, $3.57 billion in 1988 and $2.40 billion in 1987. It projects industry sales of $5.64 billion this year and $5.90 billion in 1991 (source: "PC LAN Opportunities through 1991," Salomon Brothers).

By 1992, it has been estimated that half of the 73 million installed PCs will be connected in PC LANs (Source: "1989 PC Connectivity Study," International Data Corporation).

Sold on the value of powerful computing tools positioned locally, end-users are eager to expand the LAN to embrace greater physical distances. They are beginning to realize, too, that the technology used to interconnect LANs will have to be independent of the underlying technology, able to accommodate both Ethernet networks and token ring networks in any combination—economically.

The MAN environment

In the MAN environment, cost-effectiveness can't be evaluated without considering what kind of traffic those far-flung networks will be carrying. Increasingly, it will be high-volume numerical, text and image data, applications such as scientific visualization and video communications, which demand both extremely high capacity and very high speeds. In other words, it will be digitized data traffic—bandwidth-hungry, bursty and unpredictable. And a challenge to engineer efficiently. An end-user in this environment may need only 0.5Mb/s most of the time, but may periodically run critical applications requiring 10Mb/s. That a data user would opt to buy a T3 trunk to have on hand for occasional use is hard to imagine.

The cost-effectiveness of SMDS is also tied to the way business is increasingly done. Speed is a differential advantage in today's marketplace. But the players in a business environment can change overnight, and users need the same instantaneous ability to share information with new subsidiaries and suppliers as they may have built (over time and at considerable cost) with permanent branch offices linked by private lines.

Nor can communications be limited to the intra-company associates who share private networks. Just as people keep a large number of names in their personal telephone directories because they never know whom they will need to call next, SMDS users must have the capability to share data on demand with a vast number of contacts. It often takes months of doing business before customers can determine which relationships warrant the installation of private networks. And even if cost were no object, it would be physically impractical to set up private networks on demand.

What is required is high-speed connectivity for data, and the sophisticated network management capability to run it, not only right now but in tomorrow's broadband environment as well. In this context, it's easy to understand the interest on the part of the *Regional Bell Operating Companies* (RBOCs) in SMDS as a public network service.

Cost

A public network solution reduces the cost of backbone transport because expensive facilities are shared. This represents significant savings even today, when voice accounts for 70 to 80 per cent of backbone network traffic. Within the next five years, we expect data and image traffic to exceed voice on large networks. The economics of all-private networks will likely be prohibitive in that environment.

Access costs, too, go down with a public network solution. Today's data nets are built by acquiring point-to-point and multipoint lines. The number of end and intermediary tandem points it would take to build tomorrow's mesh networks on today's model is astronomical. Not only would such a network be ferociously expensive, it would be a nightmare to manage. Network management capability will be included in SMDS, with the option for customers to manage their own MANs or to rely on the management features of the public network.

IEEE 802.6

Key to the interconnectivity of SMDS—as well as its ability to accommodate tomorrow's high transmission speeds—is that it is being built to the emerging IEEE 802.6 standard for voice, data and video. The 802.6 standard is a member of the 802 data networking family that also includes Ethernet and Token Ring technology. Some European equipment manufacturers are developing their own MAN products for the US market, and European telecommunications standards bodies are investigating the establishment of a MAN service standard.

Whether a MAN interconnects bus LANs, ring LANs, star LANs or hybrids combining LAN architectures, the basic topological elements are the same: a backbone (transport) network carrying data for all the users on the MAN, and access networks between individual LANs and the backbone network. An access loop within the access network transports signals between the user and the backbone.

Security

Security on a public MAN must, of course, be every bit as good as the security afforded by a private network. With SMDS, the *Subscriber Network Interface* (SNI), the point at which customer equipment interfaces with the access network—belongs to the customer, and only that customer's data is carried on the access network (which in essence is a dedicated path to the backbone network).

continued on next page

SMDS *(continued)*

Network administrators can configure the SMDS *screening* or *filtering* feature, which identifies a packet's designation or source address, matches it with the SNI by checking routing tables, and determines whether or not to accept the traffic. And for both public and private MANs, network administrators can do the same type of network segmentation that bridge and router configurations provide today, establishing network administration nodes as the control points.

FDDI and SMDS

Although SMDS and FDDI were designed as complementary rather than competing technologies, end-users eager to evaluate purchasing options have tended to compare them. In the near term, at least, it may be helpful to think of data network configurations in a hierarchy with today's low-speed distance-limited LANs at the bottom, private network campus environments linked by FDDI or SMDS in the middle layer and, at the top, public and private networks converging at the MAN level, with SMDS interconnecting the national services of MANs.

High-speed (above T3) data networking, though, isn't primarily a near-term challenge. Optical fiber and optical drivers already allow networking at gigabit speeds; fully optical components would permit terabit-speed networking, transmission approaching the speed of light.

BISDN

For that reason, the Regional Bell Operating Companies are positioning SMDS as a pre-broadband ISDN (BISDN) transport service, a first step in the evolution of public networks offering speeds in increments of 45Mb/s from one to 255 multiples. SMDS is compatible with but not dependent upon emerging international SONET standards for fiber-optic transmission from 51.84Mb/s to more than 13 gigabits/second. SMDS will run on both the SONET and the DS-1 and DS-3 hierarchies. That's significant for today's customers, who rarely need more than T1 or 45Mb/s capability, and who couldn't get enough SONET on premise today even if they did want higher speeds.

ATM

By 1995, though, we expect 60 per cent of the interexchange network to have been converted to SONET. As SONET becomes established as the vehicle for advanced fast-packet services such as *Asynchronous Transfer Mode* (ATM) for wide area networks and BISDN, SMDS users will realize the economies inherent in the technology. Interexchange carriers and the local telephone companies will have begun to pass along to their customers the savings resulting from decreased hardware costs in the central office (because SONET eliminates costly multiplexing and remultiplexing) as well as reduced network management costs.

A strategic question—both for end users and for service providers—is how long to wait before investing in broadband networking facilities. With the first SONET implementations not due before 1992, with ATM technology not likely to come into play until the mid-1990s, with BISDN trials several years off and *cell relay* probably not available on a large-scale national network for even longer than that, the choices come down to a fairly simple three: wait, buy now and plan to change out, or look for products featuring evolvability.

SONET-incompatible products, set at speeds below the initial OC-3 (155 Mb/s) SONET speed to be offered to customers, don't seem a practical solution. Nor do heavy investments in multigigabit ATM technology years before its full-scale implementation.

SMDS does seem to offer solutions to problems of timing on the road to BISDN. The IEEE 802.6 standards group is working extremely closely with the group developing the ATM standard for BISDN to guarantee compatibility of all basic protocols. The payload size of 48 bytes will be the same for 802.6 and ATM, as will the five-byte header size. Like ATM, IEEE 802.6 is a cell relay standard designed to support voice, data and video at 51Mb/s and up internationally. It will provide the backbone technology to be accessed through a host of standard interfaces. SONET-ready SMDS equipment—automatically upgradable to ATM hubbing—will hook a customer directly to ATM, allowing interconnection of T1 LANs, 45Mb/s LANs and SONET LANs.

Stanford SMDS trial

For these reasons, exploratory activity is heating up. Pacific Bell inaugurated SMDS in the lab last April and is now participating with Stanford University in a one-year research test. SMDS will link Stanford's medical, earth science and telecommunications centers and will transfer medical images and satellite image enhancements between LANs. Initially, all of the applications will operate at 1.544 Mb/s. Later, they will be tested at 44.736 Mb/s.

At the outset, the Stanford trial will run on an AT&T node on campus. Later it will be served by a Pacific Bell central office which will be the first fully SMDS-configured central office installation. In addition to testing advanced applications, the Stanford partnership will give Pacific Bell an opportunity to study and define customer network management elements, capabilities such as provisioning, updating, routing and real-time diagnostics. There is likely to be a range in the degree of control customers will want over their SMDS-based networks, and the Stanford trial will serve as a testbed. A commercial customer technology test is expected to begin in November of this year, to conclude in May of 1991. Product integration is scheduled for the fourth quarter of 1991, with market availability fourth quarter or first quarter of 1992.

In addition to individual initiatives such as those underway in Pacific Bell territory, the RBOCs have joined with Bellcore in an early deployment team designed to ensure that SMDS is rolled out as a fully available ubiquitous nationwide service.

LARRY HUGHES is a market and project planner with AT&T Network Systems.

STEVE STARLIPER is the project manager for SMDS at Pacific Bell in San Ramon, California. He holds an MBA in Telecommunications from Golden Gate University.

S M D S

BROADBAND SERVICE

INTEROP 90 will showcase a nationwide SMDS network, jointly developed by BellSouth, Nynex, Pacific Bell, Southwestern Bell and AT&T. Four DS1 lines will link the San Jose Convention Center through the Oakland switch with White Plains (New York), Cedar Knolls (New Jersey), St. Louis (Missouri) and Atlanta (Georgia), to demonstrate advanced multimedia, image transfer and LAN interconnect applications. Also, don't miss the SMDS conference session, S5, at 10:30am on Wednesday, October 10, and the SMDS Birds Of a Feather session on Thursday at 6pm.

A Programmer's Overview of X

by Wayne R. Dyksen and John T. Korb, Purdue University

With nearly every major workstation vendor shipping a version of the *X Window System* with their hardware, it is clear that applications written to use X will enjoy a large market. This article provides a brief overview of the facilities the X Window System and its *toolkits* provide the programmer.

Client–Server model

The X Window System is based on the *client–server* model of distributed computing in which a server coordinates the access of multiple client programs to a shared resource. In this case, the shared resource is a user's display, mouse, and keyboard, and the clients are interactive application programs.

The *X Server* allows multiple application programs to coordinate their use of the display by essentially providing each application with a "virtual display" or *window* into which it can draw text and graphics. The Server also determines which application is currently interacting with the user and passes notification of keyboard and mouse activity to that process.

Network-based protocol

The *X Protocol,* which describes the communication between an *X Client* and the X Server, uses a reliable byte stream connection (e.g., TCP) between client and server processes. The protocol describes the format of messages exchanged between client and server over this connection.

The implication of this organization is that the client and server need not be running on the same host processor. In fact, they need not be using the same programming language, hardware architecture, or operating system. They only must share a common transport protocol. A programmer can write and build an interactive application on one computing system but allow it to be accessed at any workstation supporting an X Server.

A single client application can communicate with multiple X Servers simultaneously, thus permitting a form of centrally-coordinated but distributed computing. This technique allows simple multi-user, multi-machine access to a central database without the need to implement interprocess locking protocols (since only one process, the X Client, is accessing the database).

Xlib

While the communications between an X Client and the X Server are defined by the X Protocol, programmers need not generally concern themselves with the details of these messages. Instead, the lowest-level access to the X Protocol is provided by a collection of C language interface procedures called *Xlib.*

The procedures in Xlib translate client requests to protocol requests, parse incoming messages (events, replies, and errors) from the Server, and provide several additional utility functions, such as storage management and operating system independent operations. These utility functions make writing X programs portable across a wide range of UNIX and non-UNIX systems.

The X Graphics Model

The X Protocol provides a fairly simple, hardware-independent, two-dimensional graphics model. The X Server constrains output to a typically-rectangular "window" on the display and provides coordinate mapping that makes the upper left-hand corner of the window location (0,0) in the application coordinate space. All graphics primitives are in terms of screen *pixels* on the user's display.

Thus, if the application is to draw a one inch line on the display, the programmer must query the Server to find out how many pixels there are to an inch and perform the appropriate scaling.

All graphics in X are performed in "immediate mode." That is, the operation requested (e.g., *draw line*) is performed by the Server and then forgotten. There are no display list management features currently "built in" to the X Server or X Protocol; they are to be performed by the client application. An extension to the X Server, called *PEX,* is being developed to support the 3D, display-list facilities of the *PHIGS* graphics standard.

Despite the lack of automatic coordinate transformation and display list support, the graphics model is otherwise fairly complete. In addition to the usual graphical primitives—point, line, rectangle, polygon, circle, arc, etc.—the X Protocol supports a variety of area operations that permit screening, hatching, and tiling with several bitwise Boolean operations. The color model maps well into most hardware colormaps and allows efficient implementation of "overlays" and machine-independent colormap manipulation to achieve special effects.

Exposure-based display management

Drawing graphics and text into an X window is complicated by the fact that the X Server may, at any time, generate "expose" events. These expose events announce to the application that an area of the screen that was formerly hidden has just been "exposed" or cleared and is now visible to the user. The application must then determine what image is supposed to be in the exposed region and reissue the necessary X graphics requests to redraw the area.

Window management

One of the tenets of the X Window System design was to have the Server provide "mechanism not policy." The idea was that rather than spending a lot of time arguing over policy issues such as "look and feel," the X developers would provide enough mechanism to implement a variety of interface policies.

One outcome of this philosophy was a mechanism that allows wholesale replacement of one of the dominate components of any window system: the *window manager.* The window manager is responsible for controlling the layout and organization of windows on the display. It implements the wishes of the user who, through the window manager, moves, resizes, raises, and lowers windows, as well as performing a number of other operations (e.g., deleting a window or launching a new application).

Tiles

In X, the window manager is a separate, user-level process. Like the shell in UNIX, the user can modify or replace the window manager. This mechanism has resulted in the development of and experimentation with several window managers, implementing widely different user interface styles. The most common style of window manager is one that uses overlapping windows where windows are given as much screen space as the application requests and forced to overlap or obscure one another on the display. *Tiled* window managers have also been implemented in which windows created by applications are resized and moved to "tile" the display and prevent overlapping.

Input focus

One of the responsibilities of the X Server is to decide which application is "active" and so should receive user input such as keystrokes or mouse button presses. The "input focus" is generally assigned to an application by the window manager using one of two techniques: click-to-type or mouse-to-type. With "click-to-type," the user clicks a mouse button over the window he wishes to receive the input focus.

continued on next page

A Programmer's Overview of X *(continued)*

Subsequently, regardless of where the mouse cursor is on the display, keyboard input is passed to the application(s) that has expressed an interest in that window. With "mouse-to-type," the window manager assigns the focus to the window currently under the mouse cursor. As the user moves the mouse, the focus follows the cursor.

Interclient communications conventions

A client application has a great deal of control over not only the window or windows that it creates, but also over many other aspects of the Server. For example, an application can control the focus style in use, or change the mouse acceleration parameters. Since the only contact with the Server is via client applications, such generality is necessary for the user to be able make these changes.

On the other hand, it is important for programs to be disciplined about the way in which they make changes to the user's environment. It would not do for applications to make changes—especially permanent or global changes—to the behavior of the Server without the user's consent.

To prevent such unwanted side effects, a set of conventions are documented in the *Inter-Client Communication Conventions Manual* (ICCCM). These conventions provide a kind of "mutual understanding" among clients and the Server. They describe not only the kinds of actions a client is expected to avoid, but also how clients can communicate among themselves, such as exchanging data stored in a "cut and paste" operation.

The X Toolkits

The complexity of the low-level Xlib interface and the underlying X Protocol is handled by an increasing variety of available "X Toolkits." The X Toolkits are software libraries that provide high-level facilities for implementing common user-interface "objects" such as buttons, menus, and scrollbars, as well as layout tools for organizing these objects on the display.

The basis for a family of toolkits is provided with the standard X releases from MIT. The library, called the *X Intrinsics* or *Xt,* forms the building blocks for sets of user interface objects called "widgets."

The X Toolkits come in two basic varieties, those based on the X Intrinsics and those that are not. The toolkits based on the X Intrinsics share common facilities, for example, setting parameters (or resources) and, to some extent, can be intermixed. These toolkits also generally require use of Xlib facilities to handle graphics. Toolkits not based on the X Intrinsics generally have their own graphics model and the programmer does not directly use X protocol requests via Xlib to draw graphics or text.

Widgets

For toolkits based on the X Intrinsics, a common interface mechanism called a "widget" is used. A widget is essentially an X Window plus some additional data and a set of procedures for operating on that data. Widgets are a client-side notion only; the X Server and X Protocol do not understand widgets.

Object-oriented programming style

The X Intrinsics and the widget sets based on them are built in an object-oriented style. This style provides a framework of conventions that allows code from different programmers to be easily combined into a single application without naming conflicts or problems caused by conflicting access to shared resources.

Another result of the object-oriented nature of the widget sets is that it allows them to be extended using the "subclassing" mechanism of object-oriented technology.

Resources

A typical X program, especially one based on the X Intrinsics, contains many program attributes, or "resources," that must be specified. Resources include such attributes as foreground and background colors, font, and height and width. A mechanism for organizing the assignment of values to the myriad resources in a program is available using the *Resource Manager.*

The Resource Manager is a set of procedures that allow the user, the site administrator, and the programmer to provide values for the resources used in an application. The resources are named using a hierarchical organization that identifies widgets in the application by their position in the widget tree. Since the user, or even the programmer, might not know or be concerned with every level of the tree structure, the Resource Manager allows wildcards (asterisks) to be inserted in the hierarchical specification to denote arbitrary intervening levels. It also allows widgets to be identified by the "class" to which they belong. This allows, for example, all "labels" to use a bold font or all "commands" to have a red background.

Interface Builders

Building an application by specifying in a **C** language source program all of the widgets that make up an interface, including their hierarchical relationship to other widgets can be tedious, error prone, and difficult to modify.

A solution to this problem is a special-purpose language, either textual or graphical, that allows widgets to be created and organized in a simple, direct fashion. These languages, called *User Interface Languages* (UIL), allow non-programmers to design, implement, modify, and test user interfaces. They—especially the graphical-based languages—remove much of the drudgery of constructing a complete user interface.

WAYNE DYKSEN in an associate professor of Computer Science at Purdue University. He has developed and teaches an X11-based course on interactive computer graphics. Professor Dyksen is currently testing and evaluating X11 releases for a number of vendors. He is leading a joint industry/academic research project for evaluating new X11 display station technology. He holds a Ph.D from Purdue University.

JOHN T. KORB began working on user interfaces at Xerox Corporation. Since then, he has been on the faculty of the Computer Science Department at Purdue University where he has taught courses and done research in user interfaces, operating systems, networks, and programming languages. He taught a graduate seminar in window system design, using examples from the X Window system and other contemporary window systems. Dr. Korb is the director of research facilities for the Computer Science Department at Purdue University.

The X™ Window System

Wayne Dyksen and John T. Korb will both be teaching courses on X Windows at INTEROP 90. Dr. Dyksen's class, T19, is entitled "Introduction to the X Window System," and Dr. Korb's class, T20, is called "Introduction to X Toolkits." See the INTEROP 90 final program for details. Several vendors will also be demonstrating X Window compatible systems on the exhibit floor.

Wireless networking for the 1990s

by Richard C. Allen, Photonics Corporation

Introduction

While the 1980s was known as the "Network Decade," the 1990s may very well turn out to be the "Decade of the Wireless Network." Until recently, computer communications have been constrained by the need to connect computers with cabling. Recognition of the magnitude of investment, direct and indirect, associated with network cabling has led to the development of structured cabling plans designed to help manage the costly jungle of networking infrastructure.

Why wireless?

For many applications conventional cabling isn't the best, or even a practical, solution. Users have been attracted to wireless networking for a number of reasons:

- *Quick deployment:* Cabling systems require lengthy planning and installation cycles. Once installed, cabling systems restrict an organization's ability to relocate computers or terminals to locations which have been pre-wired. This isn't always practical when office layouts change frequently. Wireless networks can be rapidly deployed without planning or installation lead-time and permit rapid response to unplanned moves and changes.

- *Freedom to move:* Structured cabling plans provide excellent economics when officer layouts are stable and the pace of change is moderate. Relocations of users within the pre-planned and pre-wired grid can be accomplished conveniently and rapidly. However, many organizations have not already pre-wired all of their desktops. Furthermore, the high rate of reorganization which characterizes today's networking world frequently requires use of temporary installations which cannot justify the high costs associated with structured cabling plans. Wireless systems allow the connection cost to be capped by permitting hardware reuse over a full life-cycle.

- *Essential applications:* Wireless connections are absolutely essential for applications using handheld, mobile computers for workers with no fixed place of work e.g., warehouse workers, factory floor controllers, workers at construction sites, field sales or service personnel and truck drivers. In the future, when wireless products support both voice and data simultaneously, we won't need wire at all for many situations.

- *New Nodes on Old Desks:* Even less mobile applications can benefit from wireless communications. These requirements are characterized as "new nodes on old desks." Here the desks are in place already, equipped with telephones, but are not wired for networks. The workers may already be in place at the desks or they may be moving to a new location which has existing desks with telephones but without data cabling. Waiting for the cabling can seriously delay projects or cut productivity.

- *Temporary Installations:* Temporary network installations, e.g., project teams, task forces and audit teams, are natural wireless applications. Sometimes temporary facilities will be leased. In both cases, the cost of installing cabling cannot be justified in view of its short useful life and the time-critical nature of typical projects does not allow for lengthy planning and installation cycles. Wireless networks can be quickly deployed to meet unplanned requirements and reused when requirements change.

- *Special situations:* Sometimes, cable installation is extremely difficult because of environmental factors such as embedded asbestos, historic buildings which cannot be gutted for cabling, cement floors which require costly cable channel sawing, buildings with already crammed cable ducts (a frequent situation), and clean rooms in semiconductor and disk drive manufacturing.

The general manufacturing environment is usually hostile to cabling. Frequent relocations of factory equipment and a high electrical noise environment raise the cost and difficulty of using conventional cabling. Current trends toward flexible manufacturing will make wireless an even better option for general manufacturing.

Cabling can be quite expensive in installations requiring some level of data security. Certain security levels require all cabling to be visible and inspected frequently. Other, higher security levels, require cable to be placed in pressurized conduit to permit detection of attempts to tap the cable. Wireless technologies based on infrared light can prevent unauthorized interception by anyone outside the secured area.

Following disasters, data processing systems must be rapidly deployed to temporary sites. These are unlikely to be friendly to cabling and the economics of cabling for temporary installations are particularly bad.

Classrooms using computers for computer-assisted instruction are often frequently reorganized as class sizes and requirements change. Wireless applications offers great flexibility in such cases.

Temporary setups for workers affected by building renovations are natural wireless applications. Even fully-cabled facilities can make use of wireless links as temporary repairs when cables fail, avoiding the lengthy downtime associated with tracking faults through a jungle of cabling.

An application for wireless networks exists in linking together separate buildings in a campus or urban environments. Lengthy delays associated with obtaining cable rights-of-way can thus be avoided.

Wireless technologies

Wireless data communications has a long history involving a diverse set of technologies:

- *Conventional Narrowband Radio:* When we think of wireless, our thoughts turn naturally to radio. Radio-based systems employing free space and power line conducted systems have been used for years. These systems use conventional narrowband radio links to provide relatively low-speed communications. Speed is limited because of the limited amount of spectrum space which has been made available.

- *Power Line:* Dating back to World War II, power lines have been used as communication links for radio amateurs. More recently, radio-based systems, operating at low frequencies, have been used for data communications in limited areas; but these systems have some problems due to unpredictable areas of coverage.

- *Point-to-Point Infrared and Microwave:* For some years, the building-to-building application has been served by directed infrared light beams produced by laser diodes or microwave links. Both methods provide high speed capability at relatively high cost.

continued on next page

Wireless networking *(continued)*

- *Unaimed Infrared:* In the late 1970s, IBM and others experimented with diffuse infrared links which flooded the room with infrared light, providing a relatively low speed communications link.

Newer technologies

Wireless technologies have been evolving rapidly in recent months. These new wireless technologies utilize either infrared light (like that produced by TV remote controls) or microwave radio frequencies.

- *Spread spectrum packet radio:* A number of suppliers have taken advantage of the FCC's approval of use of spread spectrum radio (another technology dating back to World War II) in certain microwave bands. Spread spectrum radio provides strong benefits through its ability to share spectrum space while minimizing interference and the probability of interception by unauthorized recipients.

- *ARDIS:* Motorola and IBM announced a joint venture called ARDIS which provides a nationwide network of medium speed communications links with "bandwidth on demand" for portable applications like field service.

- *Cellular:* The rapid proliferation of cellular radio telephone systems has led to marketing of modems optimized for the cellular environment. This makes possible operation of your computer from your car—if you are brave enough.

- *Packet radio:* Packet radio, developed for military applications has been applied to both narrowband and spread spectrum techniques. Radio amateurs developed protocols which have found application in both environments.

- *Directed infrared:* Earlier undirected infrared wireless links suffered from poor power efficiency and low speeds. By directing the infrared light into narrower beams, pointed at small areas of the office walls or ceilings or to ceiling-mounted repeaters, operation at network speeds of up to 10 Million bits/second has been achieved.

- *Microwave in-building:* In the spring of 1990, Motorola received permission from the FCC to use microwave frequencies for wireless-in-building applications. This system will operate at 18GHz and will permit 10 Million bits/second operation.

All of these technologies are relatively new to the commercial marketplace. Since they have complementary characteristics, it is likely that each will find application areas matching its particular strengths. Infrared solutions seem particularly useful in today's open offices and flexible manufacturing environments. Solutions employing microwave radio frequencies seem particularly well suited to fully mobile applications and indoor use in situations where all offices are fully enclosed by floor-to-ceiling walls.

Public acceptance of infrared is helped by familiarity with the home TV remote control. The acceptance of microwave radio frequency technologies will be helped by the growing use of cellular telephone systems. Growth of radio technologies may be hindered, however, by the limited frequency spectrum space available for such use and worker fears of possible effects of exposure to microwaves.

Technology characteristics

Obviously, each of these wireless technologies has its own characteristics which make it more suitable for some applications than others. Table 1 summarizes some of the more important characteristics of current wireless systems. With the pace of innovation exhibited in recent months, this table is destined to rapid obsolescence.

	Typical Speed	*Security*	*Range*	*Typical Cost per Node*	*Deployment*	*Interference*
Narrowband Radio	to 4800 b/s	Insecure	National coverage of metro areas using VANs or 3–10 miles local range	$2000 – $3000 plus VAN fees (if used)	Rapid once license is obtained	Interference is a concern
Spread Spectrum Radio	38.4 Kb/s to 250 Kb/s	Moderately secure	300 to 500 ft indoors; 1 Mile outdoors	$300 to $2500	Rapid. No license necessary.	Interference resistant Near-far problem
Cellular	2400 b/s to 16.8Kb/s	Insecure	Unlimited using cellular network	$1000 to $1300	Rapid	Interference resistant
Directed Infrared	230 Kb/s to 10 Mb/s	Secure	100 ft	$250	Rapid	Interference immune
Microwave	to 10 Mb/s	Insecure	Building coverage	Not announced	Unknown	Exclusive frequency assignments limit interference

Table 1: Characteristics of wireless technologies

Summary

Network cabling systems have been enthusiastically adopted by large organizations. Nevertheless, these do not meet *all* of an organization's needs. Where cabling is inconvenient, untimely or impossible, todays users have a wide variety of complementary wireless technology choices. The increasing awareness by network users and administrators of the wireless option will lead to the 1990s becoming the "Decade of the Wireless Network."

RICHARD C. ALLEN is the president and founder of Photonics Corporation of Campbell, California. A veteran engineer with top management experience at NCR, Memorex and Varian, Mr. Allen has just completed four years of research and development on an infrared-based alternative to traditional cabling solutions in computer and terminal connectivity. He holds B.S in Electrical Engineering from Rutgers University.

Learn more about this topic at INTEROP 90! Attend S25: "Cableless Networking" on Thursday October 11, at 1:30pm.

A Brief Perspective on Gigabit Networking

by Craig Partridge, BBN Systems and Technologies

Introduction

Gigabit networking is a relatively new field. However, research work on this topic is already quite vigorous (over 75 papers so far this year alone), and it is beginning to be possible to talk intelligently about the general scope of gigabit networking activities. In this article, I attempt to present a broad view of the current state of gigabit networking research.

A new approach to networking: cells

One of the major ideas in gigabit networking is that future networks may use fixed length packets, called *cells,* as the basic unit of communication. One particular form of cell networking, called *Broadband ISDN* (BISDN), is being developed by ANSI T1S1 and CCITT as an international standard. BISDN uses cells which are 53-octets long.

Why use cells? There are a variety of reasons. One is the realization that ISDN, as it currently is designed, doesn't scale well to high speeds. Consider the current model of ISDN: a subscriber gets a link with a certain amount of bandwidth, which can be dynamically subdivided a limited number of ways. Consider, for example, the basic rate ISDN service of 2B+D channels (2 64Kbps B channels plus 1 D control channel). Now take an application that wants to send about 40Kbps second. It must allocate one of the 64Kbps B channels, and waste about 24Kbps bandwidth to get the bandwidth it needs. At a gigabit this wasteage would get much worse. (Consider the same 40Kbps application, forced to allocate a 1 megabit channel because 1,000 1 megabit channels was the smallest way a gigabit ISDN channel could be divided). The problem is that ISDN is using bit multiplexing to subdivide the link bandwidth, and that multiplexing can only be done a limited number of ways.

Enter cells. If cells are used as the unit of multiplexing, then a gigabit channel can deliver millions of cells a second. Furthermore, each cell stands alone, as a distinct unit. So a host at the end of a BISDN channel can allocate those millions of cells a second among its applications in any way it wishes. Thus cells offer much greater flexibility.

From a networking point of view, cells are also interesting because we actually have a certain amount of experience with them: in the form of slotted local networks such as the *Cambridge Ring,* and in high speed digital switches, such as AT&T's *Starlite* and IBM's *Paris.* Researchers believe they actually know enough to try to build an internetwork of cell networks, using these components. Furthermore, work is already underway to standardize the first layer of protocols that would run over these networks: the so-called *Segmentation and Reassembly (SAR) Layer.* The purpose of the SAR is to fragment data units into cell-sized chunks at the sender, and reassemble the cells back into larger data units at the receiver.

Current protocols at gigabit speeds

While there's considerable interest in cell networking as a possible architecture for gigabit networks, there's also some work being done to see if current protocols can go at gigabit speeds. The pleasant (and somewhat surprising) answer is apparently yes, current protocols *can* go at gigabit speeds.

A study by Dave Clark, Van Jacobson, John Romkey and Howard Salwen [3] suggests that, on the faster processors we expect to have when gigabit networks arrive, the protocol processing required for gigabit speed TCP/IP will be achievable.

A similar analysis for IP routers, has shown that, in theory, such routers could be built from today's faster processors and achieve near gigabit throughput.

The results of these surveys are also applicable to OSI's *Connection - less Network Service.* No study, to my knowledge, has been done on the feasibility of scaling up the OSI *Connection Oriented Network Service* to gigabit rates.

Applications

The reader might reasonably wonder why, if we think current protocols will go at gigabit speeds, are we considering a new networking approach like cells? The answer is that while we could scale up existing protocols, there's reason to believe the kind of networking service that current protocols would provide are not the kind of networking service we want on gigabit networks.

For example, one major application that gigabit networks will make possible is high-quality integrated voice and video (multimedia conferencing). Voice and video conferencing has some strict limits on the variation in network delay. If the delay gets too variable, the voice and video quality will deteriorate. Now imagine the following scenario: You're sitting in your office having a multimedia conference with a couple of customers. Then the guy in the next office fires up a big file transfer, using very large datagram sizes. His large datagrams get mixed with your conferencing datagrams at your local router, thus every so often, one of your datagrams is delayed by one of his. The voice quality in your conference gets so bad, you have to end it early.

The problem here is that variable-size, and potentially large, datagrams can cause large variations in delivery times for other users's traffic. There are researchers who believe they have solved this problem for large datagrams, but others aren't so sure. By using fixed packet sizes, cell networks may make the problem of variation in delay more tractable.

Another reason to be interested in cell networks is that cells, again because of their fixed size, make it easier to build parallel switching systems. There's some expectation that putting parallelism into switches will make it easier to scale them up from gigabit to multigigabit speeds.

Many of these issues will be better understood after researchers have had a chance to experiment on the gigabit testbeds being installed this year. You should expect a number of issues to be resolved (and some new ones discovered) as a result of experimentation over the next few years.

References

[1] C. Partridge, Ed., "Workshop Report: Internet Research Steering Group Workshop on Very-High-Speed Networks," RFC 1152.

[2] A. R. Jacob, "A Survey of Fast Packet Switches," *Computer Communication Review,* January 1990.

[3] D. D. Clark, V. Jacobson, J. Romkey, and H. Salwen, "An Analysis of TCP Processing Overhead," *IEEE Communications Magazine,* June 1989.

Learn more about gigabit networking at INTEROP 90. Attend S10; "Progress in Gigabit Networking," Wednesday at 1:30pm.

CRAIG PARTRIDGE received his B.A. (1983) and M.Sc. (1988) from Harvard University and expects to get his Ph.D. from Harvard any day now. For the past seven years he has worked for Bolt Beranek and Newman on a variety of networking related projects including CSNET, the NSF Network Service Center (NNSC), and various projects concerned with distributed systems, IP transport protocols, network management and gigabit networking. In addition, he is a member of the Internet End-To-End Research Group, the Internet Engineering Task Force, and the Internet Engineering Steering Group.

The OSI/Network Management Forum: Achievements and Objectives

by Bruce Murrill, OSI/Network Management Forum

Introduction

This article identifies the key deliverables produced, current work areas being addressed, initiatives taken and future plans for the *OSI/Network Management Forum* (OSI/NM). It also outlines the steps that users of network management should take to identify a plan for the introduction of open, interoperable systems into the management of their information environments.

The OSI/Network Management Forum was created in July 1988 to accelerate the development and use of OSI standards for network management applications. Its members expend considerable resources toward achieving Forum goals. They realize, however, that unless the users of network management products and services understand the Forum's work, the widespread implementation of open network management systems may never be achieved.

Networking in the 1990s

In many parts of the world, the marketplace for telecommunications is becoming deregulated. In Europe, North America and Asia, many countries have already taken steps to open the market for equipment and services used by customers to serve their telecommunications needs. Many other countries will soon follow. The most noticeable effect of this change in market structure is that many suppliers are entering the market, providing new technologies and new applications to serve customers' needs.

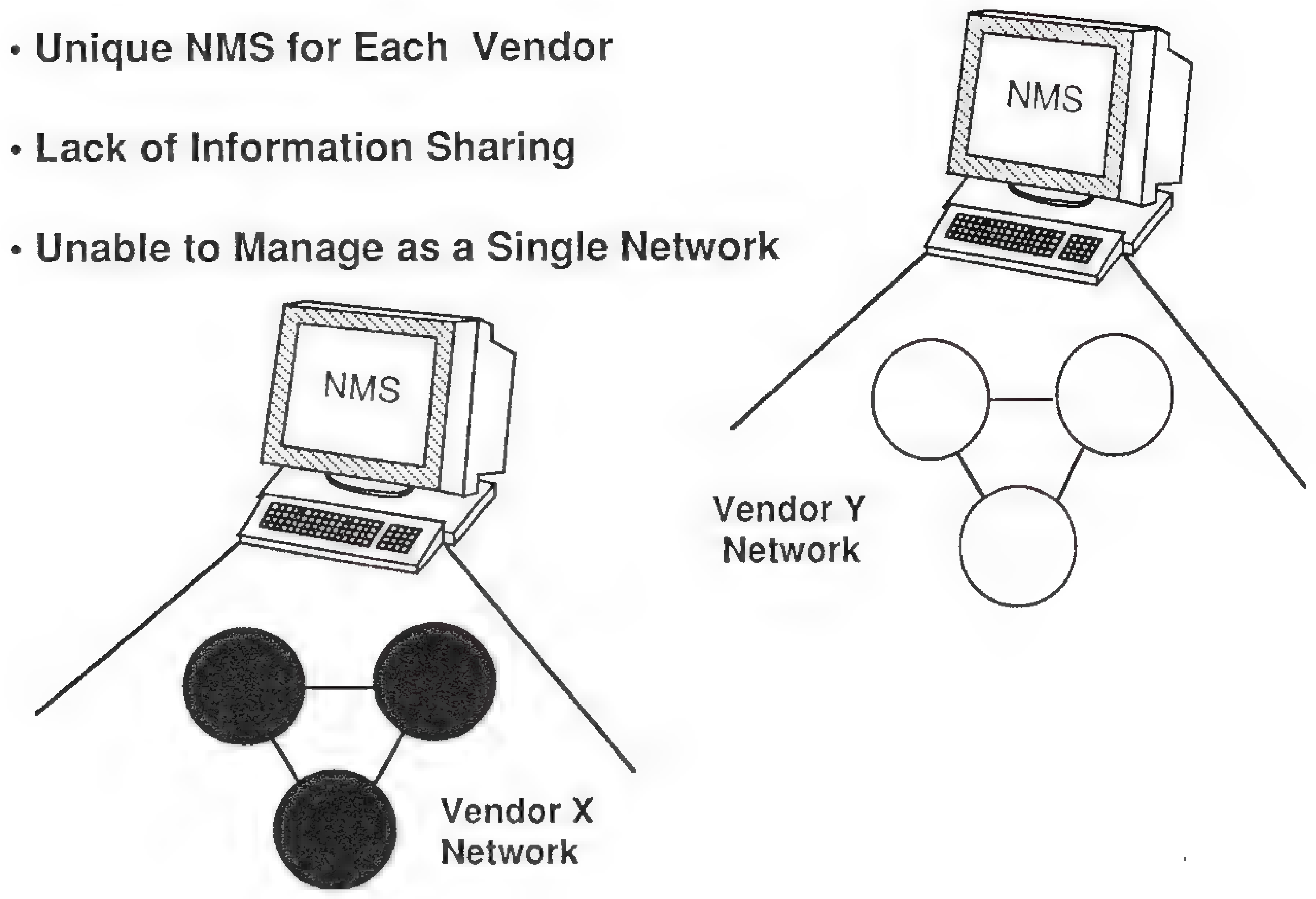
In the United States, the average private network is comprised of equipment and services from some 20 suppliers. Some global private networks number suppliers in the hundreds.

The advantage to the customer of this highly competitive environment is a wide diversity of choice, and the opportunity to demand ever greater capabilities from their suppliers. Managers of large corporate and government networks have reaped the benefits of increased network performance and added network control capabilities. They have demonstrated the strategic importance of moving towards a full-functioning, well-managed network to serve their companies' voice and data needs.

However, with the recognition of a network's strategic importance has come increased demand for its uninterrupted availability. Along with choice of suppliers has come the complexity of managing, as a single entity or resource, what has become a network without boundaries.

Complex networks now cross geographic boundaries, as more and more companies adopt a "global view" of their operations. They cross traditional "voice" and "data" boundaries as companies begin to manage their entire range of applications as part of a single entity. They cross supplier boundaries as customers choose the best mix of products and services to meet their needs.

Many of today's network components come with a *monitoring* and/or *management* capability. In fact, most customers have come to demand management features when making their buying decisions, and vendors have been quick to comply. However, the management capability is usually proprietary or unique to the individual network component, or sub-network provided by that vendor. There is no good way to tie the various systems together into a single, manageable whole.



- Unique NMS for Each Vendor
- Lack of Information Sharing
- Unable to Manage as a Single Network

Figure 1: The Network Management Dilemma

Multi-vendor Network Management

The network managers of the '90s are demanding that the network management capabilities provided by their different suppliers fit into their multi-vendor network configuration. They expect sufficient flexibility in the management systems so that they can customize a network management environment for their own unique operation. Making disparate systems work together as a single entity under the control of the customer requires a common approach by the various suppliers, and agreement to employ a common interface for the exchange of data. While it is possible to connect systems together using several proprietary protocols, use of standard protocols and application services, with common definitions of terms, makes for a more cost-effective, stable, and reliable approach.

Prior to the Forum being formed, there was no common approach, and interfaces among systems were largely proprietary. Some suppliers had attempted to base their interface specifications on the *Open Systems Interconnection* (OSI) standards work in progress, yet even when those vendors compared their approaches, there were obvious differences. Some of the differences were the result of industry fragmentation, such that individual segments of the market took differing views regarding standards development. Even within an industry segment, such as telecommunications, the standards allowed a wide range of options.

Given the number of possible combinations of options available, it was clear that suppliers would not make the same implementation choices, and would not, therefore, be able to interconnect their systems without building a customized interface.

The supplier community could have chosen to wait until implementation profiles for network management were defined in the normal course of standards development. However, the pressing need for real open system products in network management, and the desire to reach agreement *before* each supplier developed an entrenched position, led to the Forum's creation.

continued on next page

OSI/Network Management Forum *(continued)*

The Forum

The OSI/Network Management Forum is an international group of companies who have a stake in the development of network management products and services. It is a pragmatic group that crosses the boundaries of computing and telecommunications, service providers, vendors and users.

The Forum's mission is to accelerate the development and use of international standards for network management. That mission is directly aimed at benefiting the user by providing the means of managing multi-vendor, multi-service networks. Fundamental to the Forum's mission is the objective of achieving interoperability between disparate network management systems to meet today's needs in the marketplace and in the timeframes required by the user. The idea is to define a single implementation of the emerging standards that can be used as the definition of a common network management interface that will allow their products to interoperate.

Single implementation of standards

The main activity of the Forum is the definition of this single implementation of the standards. This consists of sifting through the many allowed options, and selecting those that best suit a network management application. This activity is critical to the developers of network management systems, for without a complete, detailed specification, different developers will make different choices, and true interoperability will remain just out of reach. This work is made more difficult in areas where the standards are not yet defined. In those cases, the Forum works from draft standards, if available, and makes informed judgements, where they are not, in order that the specifications will be complete and usable.

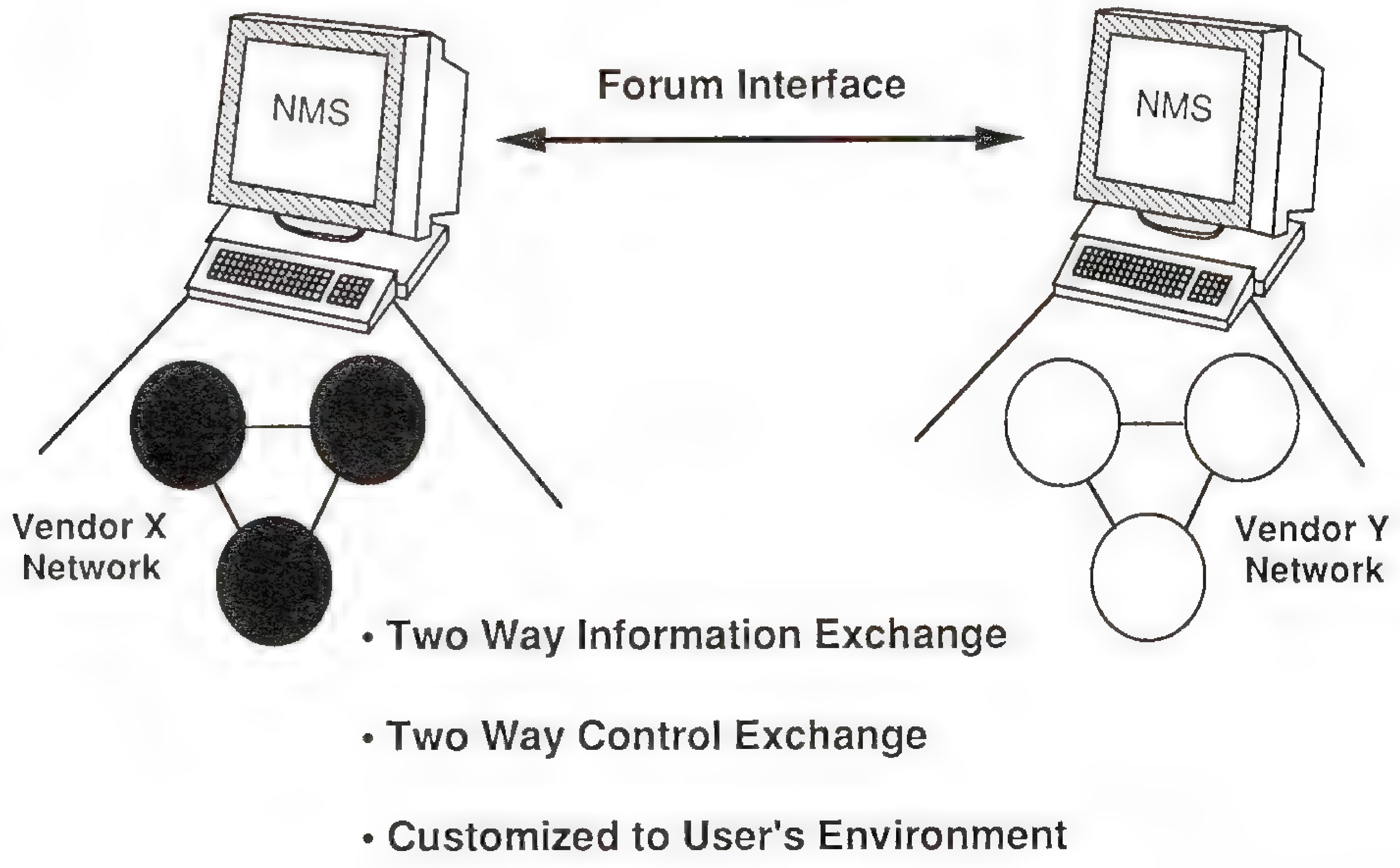

NMS
Forum Interface
NMS
Vendor X Network
Vendor Y Network
• Two Way Information Exchange
• Two Way Control Exchange
• Customized to User's Environment

Figure 2: The Forum's Solution: Interoperability

The specifications work is only the necessary first step. What is important to the users of network management products and services is seeing real products that can exchange network management information to support new, integrated applications. An equally important part of the Forum's work program, then, is to promote and encourage the use of the Forum's work, as early as possible.

To accomplish that goal, the Forum works with other organizations to speed the support capabilities, such as conformance testing, that are necessary for product introduction. The Forum promotes the work of its members, in order that users can see the real application benefits of the Forum's work. It also provides overall education to the users about what they should expect of a product or service that conforms to Forum specifications.

The Forum does *not* create standards. It encourages the standards bodies around the world to *move quickly* and to adopt a *common approach*. Neither does the Forum develop products. Product development is left to its members, and the Forum simply provides opportunities for members to show their products. The Forum is a catalyst—a catalyst for change—to make things happen quickly in a common way.

Membership classes

The Forum works as an open consortium of members from around the world, and its work is accomplished through the efforts of the members. Two types of membership are offered. *Voting members* commit working capital and technical people to fuel the Forum's work programmes. *Associate members,* which make up the bulk of our membership, pay annual dues and may contribute to our technical work if they so desire. The Forum is run like a corporation, with well-defined milestones for delivery of the specifications. It has a Board of Trustees, operational managers to head up major work programs, and team leaders to manage the technical activities.

The Forum is very much market-driven, which explains its urgency in reaching agreement on a common approach to network management interoperability. Thus it is most interested in understanding the needs of users, so that its work programs help to meet those needs.

Forum programs

There are four main programs in progress today. Taken together, they form a comprehensive way for the Forum to accomplish its goals of accelerating the development and use of a common, interoperable interface for network management:

- *The Technical Program:* The technical program consumes most of the Forum's resources. Its mission is to deliver specifications that define a common implementation approach; an architecture; a data communications protocol; application services and object definitions—everything needed by a developer to implement the Forum's interoperable interface.

Management systems interoperability

The Forum's focus is on interoperability, the ability for different management "islands" to exchange pertinent network management information in order that the user can manage his network as a single, end-to-end entity. The Forum is concentrating on the exchange of data among management systems, regardless of the type of network any particular management system is managing. In other words, the Forum's specifications can be used to tie together the management systems of many different networks whether they be OSI networks or proprietary networks. By concentrating on communication among management systems, using OSI Management techniques, users can begin to experience the benefits of OSI right away.

Key to the concept is two-way information exchange and control. In a multi-vendor environment, this allows the user to designate which system or systems will provide a control (Manager) function, and which systems will act as support (Agent) points.

continued on next page

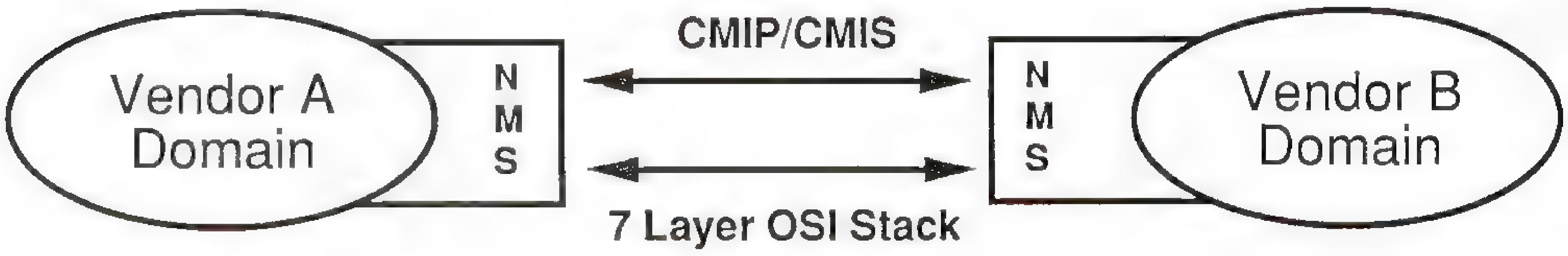
OSI/Network Management Forum *(continued)*

Interoperability is not an easy goal, particularly in an application as complex as network management. To work properly, it requires many things, starting with a flexible architectural approach. This allows many different types and levels of network management systems to be interconnected, and provides for the widest possible range of product implementation choices.

A common data communications protocol is defined as the method to be used for actual transmission of data between systems. The data itself is constructed of commonly understood resources, actions and behaviours, known as *managed object definitions,* formed into *message sets,* which are specific to network management application services. Conformance testing assures that one supplier's implementation of the Forum's specification matches another's, so that interoperability can be accomplished with minimum overhead.

The broad term "network management" encompasses a variety of functional areas. The *International Organization for Standardization* (ISO) has defined five broad management areas, each of which encompasses a multitude of tasks. The Forum's eventual goal is to address all five of these functional areas.

The protocol, the application services or management information, and the object definitions combine to provide a complete interoperable interface. Each management system manages its own network, which could be an OSI network or a proprietary network. What ties the networks together into a single whole is the ability to exchange management information in a common, consistent way.



- Widespread Usage Among Communications and Computer Vendors
- Allows Users to Mix and Match From Different Vendors
- Simplifies Interconnection

Figure 3: Benefits of Standard Protocols

Release 1

The Forum's technical specifications are being developed using a phased approach. *Release I,* now nearing completion, establishes basic interoperability between management systems. Thus, it includes some of the building blocks, such as data communications protocols, architecture, and object specification framework. These building block elements will form the basis of all the Forum Releases.

In January 1989, just six months after the Forum was created, its Protocol Specification was approved by the members and released. It defines both a wide-area and a local-area network implementation, specially optimised and tuned for use in network management applications.

Application Services (message sets) were released in June 1989 to address fault management and configuration management. These are considered by nearly all users to be the most urgent of the functional areas. Within the area of fault management, for example, the specifications provide a means of sending an alarm in a common format, such that its severity is understood by both the sending and receiving system.

The *Object Specification Framework,* released in October 1989, provides a template for use in defining objects. The Forum is using this template to create an *Object Library* of common objects, including objects that are managed, such as a piece of equipment or a communications path, and objects that are employed in the management of networks, such as a sieve. The Objects Library document, containing the agreed definitions of managed objects for Release 1 was approved in March 1990.

The *Architecture and Glossary,* published in January 1990, combines the work of ISO, CCITT and other bodies into a common, flexible architecture, and provides a unique definition of terms, so that everyone in the industry can use a single term to mean a single thing for network management.

In addition to the above, work on the *Implementation Conformance Statements for Protocols, Messages and Objects* was completed in May 1990. The Conformance Statements identify in detail the mandatory and optional implementation and testing values, ranges and constraints that should be adhered to in the various protocol, message and object data fields. This ensures that tested conformant systems have the best possible opportunity of interoperating.

Other work areas addressed in Release 1 are definitions for *Shared Management Knowledge;* how one management system understands the capabilities and content of another: and a *Naming and Addressing* definition for managed object instances; how a system unambiguously identifies that element which is being managed.

Release 2

The Forum's objective is to make one release per year, adding increased functional richness to its specifications over a wider range of agreed managed objects. Release 2 work is currently underway and addresses *Diagnostic and Testing Management Messages,* together with an update to the protocol specification for File Transfer (FTAM) to the *International Standardised Profile* (ISP). This will allow the bulk transfer of test logs between systems for analysis.

Release 3

Release 3 will cover areas of *Security of Management,* basic access controls by function/objects: *Performance Management,* objects statistics and history file requirements: *Path Tracing,* for fault isolation: *Trouble Ticketing,* objects and event reporting: *Accounting,* basic usage records for accounting purposes. All of these areas have been defined as high priority items by network management users.

As stated earlier, the Forum is committed to implementing ISO agreed standards, where available. One of our major roles in the next few years will be to provide stability for meaningful and lasting implementation, while defining a well signposted migration plan to intercept the final agreed ISO standards. One way of assisting this process is to work with the standards makers during our specification processes, to ensure that even from the beginning our specifications model the principles and much of the detail of the final standards.

continued on next page

OSI/Network Management Forum *(continued)*

Harmonisation

This process is in place with many of the regional standards bodies and the Forum can aid early harmonisation of agreements. Important areas this year, for early harmonization, are the agreement of an *International Standards Profile for Network Management* and international agreement of a stable *Guideline for the Definition of Managed Objects.* Equally important is to set up an *International Management Information Library.* This would contain a library of agreed managed objects and associated information which can be used by many different implementors in their management products.

- *Conformance Program:* The Forum's conformance program is of crucial importance if the customer is to be assured that a given management system has actually implemented the specification correctly. Because it is so important that conformance testing be conducted by an independent third-party, the Forum is working with two of the principal testing organisations in the world, the *Corporation for Open Systems International* (COS) in North America, and the *Standards Promotion and Application Group* (SPAG) in Europe.

Executive Council

In May 1989, the Forum reached agreement with COS and SPAG to form an Executive Council consisting of members of the three organizations. Its purpose is to identify those areas where the three organisations can work together to achieve faster standards implementation for network management. The conformance programme was seen by the Executive Council as the first of many areas where cooperation will have a positive effect on our shared mission.

The Forum is working with both organisations to ensure that conformance tools are available to test its specifications. Early releases of these tools were available to Forum members in mid 1990.

- *President's Roundtable:* The Forum was created to serve the user, and welcomes users as members of the organisation. However, many users, who are interested in what the Forum is doing and who wish to provide their views of our work, have neither the time nor the resources to become Associate members of the Forum. For that reason, the President's Roundtable was created. It is a program designed for users of network management products and services as a way to provide direct input to the Forum's Board of Trustees.

Generally, President's Roundtables are held just before meetings of the Board of Trustees. These are working sessions designed to impart information about the Forum's current work program, and solicit feedback on our future courses of action. Follow-up to the half-day meeting includes a free subscription to the Forum's bimonthly newsletter and member prices on Forum documentation. Participants are expected to keep the Forum informed of their needs by responding to periodic polls. The Roundtable program is active in all parts of the world, with meetings scheduled through the remainder of 1990 and beyond.

- *Network Management Showcase:* The final major program of the Forum is its Network Management Showcase. The concept of Showcase is to allow Forum member companies to demonstrate, in a structured environment, the progress they are making toward development of network management products and services using the Forum's specifications. The Showcase is not a stand-alone show, but a series of displays within existing tradeshows, such as INTEROP 90.

Through its Showcase program, the Forum acts as a catalyst to encourage member companies to develop products and services as quickly as possible. This is yet another example of how it is bringing the industry together to provide network management solutions to the customer.

Summary

In summary, the Forum now includes among its members most of the world's major information systems providers. It is making rapid progress and has had a pronounced effect on the industry. It is bringing real, tangible solutions to customers' needs in managing complex networks for the 1990s.

The Forum exists because its members believe its work is important. Individually, the member companies are investing a large percentage of their development money to reach industry agreements, and to implement those agreements. In the end, however, they will only continue on their current course if their customers demonstrate, through their purchase decisions, that open systems are of value.

It is therefore important that network management users identify a need for open management environments. It is the users who create the demand that will fuel product development by vendors.

Questions to ask vendors

Listed below are the type of questions that should be asked of vendors who have network management products. At this time OSI management may not cover all user requirements. However, the replies will give users a basis on which to plan their evolution to open management and reinforce the requirement with vendor companies.

- *What ISO defined functionality, or subset of functionality, do you support across an OSI/NM Forum compliant interface (i.e., Configuration Management, Fault Management, etc.)?*
- *What role does your management system support at the interface: manager, agent or both?*
- *What OSI/NM Forum agreed managed objects definitions are supported in your management system?*
- *Have you conformance tested your management product? If not, when do you intend to do so?*
- *What precise testing (from above) was undertaken and are test results available for scrutiny?*
- *What evidence do you have that successful interoperability testing has taken place over the OSI/NM Forum interface between your management product and another vendor's product? For what functions and objects, and in what role.*

This article was originally presented as a paper at the *Networks '90* Exhibition in Birmingham, England in June, 1990.

OSI/NM FORUM

BRUCE MURRILL was a prime mover in setting up British Telecommunications' Communication Facilities Management Division. For the past seven years, he has had responsibility for Project Management, Development and Support for Network Management Systems for private networks. Previously he was involved in software development for major communications systems. He has been the Program Director for the OSI/NM Forum for the past eighteen months.

Network management is featured in several conference and BOF sessions at INTEROP 90. See your program for details. Also, look for network management demos in the exhibition hall. A Special Session, "SNMP and the OSI/NM Forum—A Single Perspective," is scheduled for Thursday at 5pm.

DECnet/OSI Phase V: *Real OSI or Only Selected Interfaces?*

by Carl Malamud

Product marketing in many computer companies these days have a single design requirement for their engineers: "let us claim that this product is compliant with OSI." We're seeing products that comply with standards that don't exist, products that "supplement" standards, and even products that don't exist that supplement standards that don't exist.

DECnet/OSI

Digital Equipment Corporation's (DEC) latest networking products ought to be a marketing department's dream: the word "OSI" is not only present, it has been integrated into the product name. We no longer have DECnet, we have *DECnet/OSI.*

So what exactly *is* DECnet/OSI? What are the prospects for integrating DECnet systems with other vendor's computers? Did DEC do only a subset of the standards? Have the OSI protocols been integrated into the architecture or just into the product names?

First, a caveat. DECnet/OSI, Phase V of DECnet, is a *new* offering. This means that many of the pieces are just becoming clear. Most importantly, there are pieces, as with any architecture, that have not been defined, or that have been defined and will not be used. This article thus discusses DECnet as the Digital architects seem to have envisioned, not as the engineers may have implemented.

History

Let's start with a little history. DEC has gone through five stages of the *Digital Network Architecture* (DNA is the architecture, DECnet is the implementation of that architecture). Phase I was two PDPs with a wire strung between them—not exactly a network by current standards!

The current products are compliant with Phase IV of DNA. This is a network centered on Ethernet workgroups, strung together with either wide-area bridges or routers. The wide-area systems typically use DEC's DDCMP data link protocols. Built on top of the data link are proprietary network, transport, and session layers.

Phase IV of DNA doesn't have an explicit presentation layer—that function is up to the individual applications. Since most DECnet systems are VAX computers running VMS, at least for the transfer syntax function of the presentation layer, there was really no need for a separate layer in the early days.

Built on top of the DEC session layer are a wide variety of applications. The two most-used applications are DAP and CTERM. The *Data Access Protoco*l (DAP) is a record-access service, allowing access to remote files, file attributes, and a variety of record level operations such as searches by index keys. CTERM, the *Command Terminal* protocols, are used for remote interactive access. CTERM, also known as the "SET HOST" command, allows a remote terminal to appear as a local one to the host system, much like Telnet does in the TCP/IP world.

Over the years, DAP and CTERM were supplemented by a wide variety of other services. Messaging, for example, was first done using the *mail-11* message handling protocols, and lately using DEC's proprietary *MAILbus.*

Videotex, remote consoles for network management, booting of diskless nodes, distributed bulletin boards, and a host of other services are available for DECnet nodes.

DECnet Phase IV

To see what DECnet Phase V encompasses, let's start at the bottom of the Phase IV protocol stack. The subnetworks (layers 1, 2, and some of the network Layer) in Phase IV included Ethernet, the IEEE 802.3 version of Ethernet, and DDCMP for wide-area communications. In addition, Phase IV supports X.25 networks, but only with permanent virtual circuits. Finally, the IEEE token-passing bus is also supported for MAP/TOP networks.

Phase V subnetwork support

Phase V has dramatically increased the support for subnetwork technologies. In addition to DDCMP, DEC has added the HDLC protocols. In addition to IEEE 802.3, DEC has also added the FDDI services (actually, since both use the *Logical Link Control* [LLC] services of 802.2, this is not really that surprising).

More importantly, DEC has added two proprietary services that allow public data networks to be incorporated into a DEC environment. First, the *Modem Control Protocol,* a physical layer service, allows a modem service, such as V.25 or the Hayes AT command set, to be dynamically established by a higher layer. The higher layer, in this case the Network Layer, has a *Dynamically Established Datalink* (DED) component. The DED service will set up a modem call, an X.25 switched circuit or any other dynamic link on demand. It will then keep that circuit up for a management-determined period of time in case any more traffic comes through. After a period of inactivity, it brings the circuit down.

Thus, for data links, DEC supports HDLC, including dynamic X.25 or V.25 circuits. For LANs, DEC supports the IEEE LLC. This in turn, makes it possible for DECnet to easily incorporate Ethernet, FDDI, and, who knows, maybe even Token Ring.

Network Layer

At the Network Layer, there are really two issues that need to be addressed. The simplest issue is whether any two nodes will be able to read each others packet. ISO 8473 defines a standard format for connectionless data and error packets which DECnet/OSI uses.

The second issue is how nodes find out about each other. Networks are composed of *end systems* (ES) and *intermediate systems* (IS). If two end nodes on the same subnetwork want to communicate, the question of finding each other is not too difficult. The difficult issue is when an intermediate system needs to be used to route packets between different subnetworks (or within a single very large subnetwork). The question of routing information is addressed at two levels.

ES–IS

First, ISO defines an *ES-IS routing exchange protocol* which allows end systems and intermediate systems sharing a common subnetwork to find each other. This is accomplished through ES Hello and IS Hello packets which are multicast on an ISO-defined address. DECnet/OSI uses this protocol, defined in ISO 9542.

The more difficult question for routing information is dynamic exchange of routing information between different IS systems. ISO 9542 defines how an ES tells an IS about it's existence. Currently, there is no ISO protocol for IS systems to tell other IS systems about the end systems they can service.

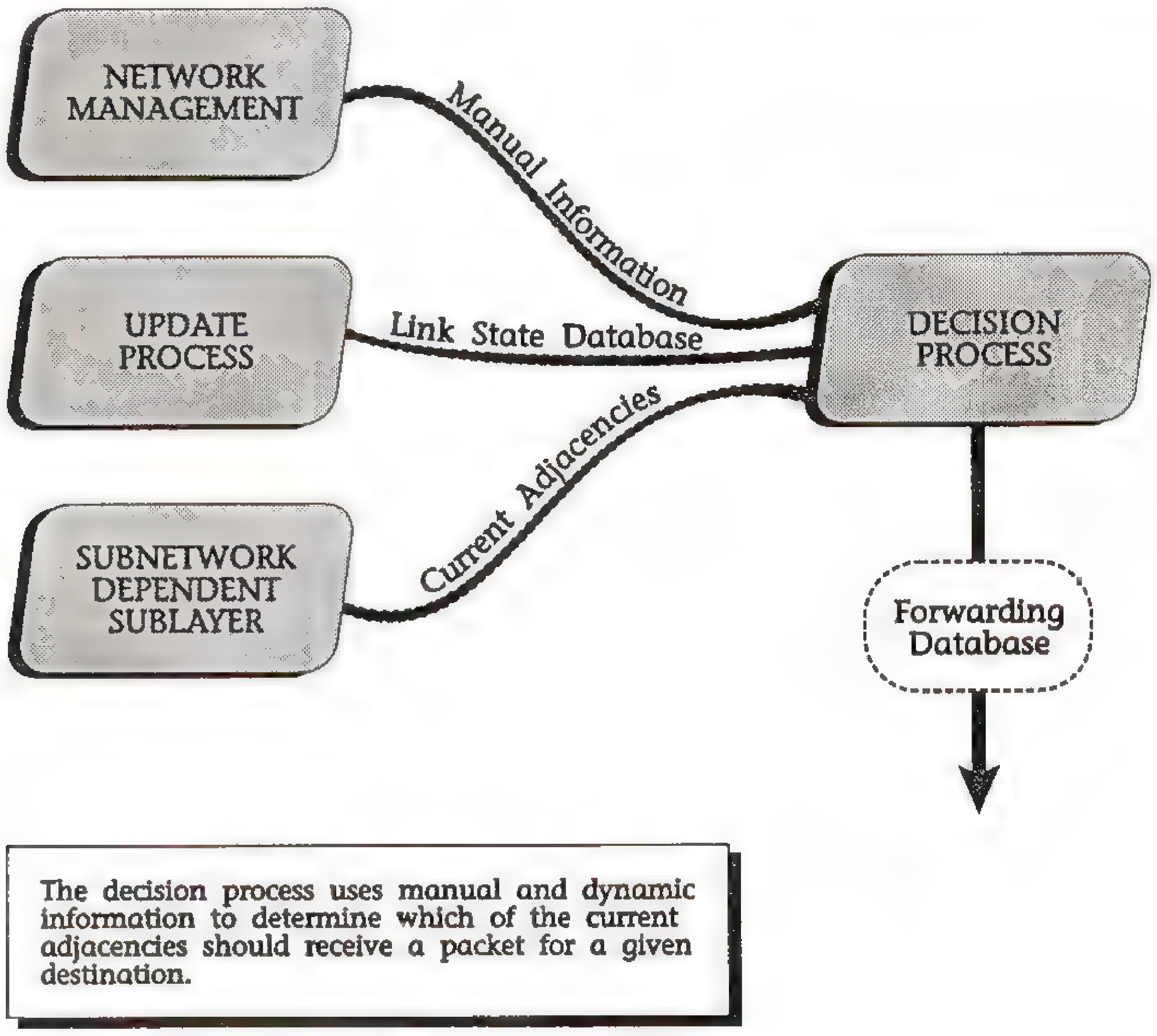
continued on next page

DECnet/OSI Phase V *(continued)*

IS–IS

The protocol that DEC uses for this information is the *Link State Packet IS-IS Protocol,* which forms the basis for the current ISO draft. DEC's IS-IS allows two routers two dynamically exchange routing information.

This is accomplished through the exchange of link state packets. A link state packet is issued by each intermediate system, and contains all of its adjacent neighbors. The link state packet is then sent to every intermediate system on the network. The link state database thus has every IS, and all links between that IS and its neighbors. In addition to the link state database, every IS also keeps two other kinds of information (See Figure 1). First, there is static routing information, entered by network management. Second, there is a database of adjacencies, compiled from the ISO 9542 Hello packets that have come in.



The decision process uses manual and dynamic information to determine which of the current adjacencies should receive a packet for a given destination.

Figure 1: Network Layer decision process

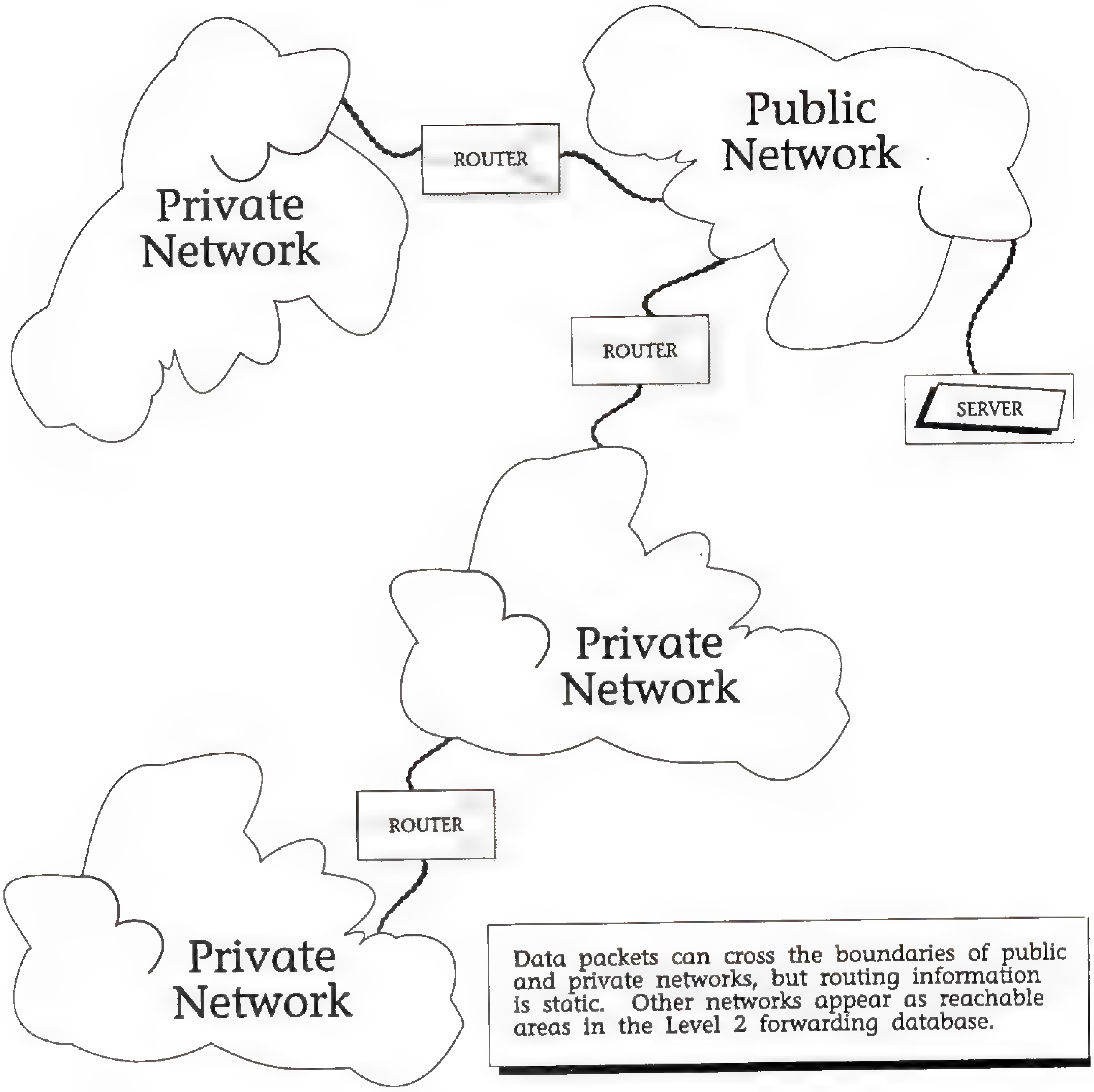
Areas

Given these three sources of information, the routing decision process is able to decide which of the adjacent neighbors would be able to move a packet for a given destination one hop closer to its destination. In order to simplify the routing decision, DEC segments networks into *areas.* Within an area, the routing decision is made on the individual node address, and is known as level 1 routing. Outside an area, the routing decision is made solely on the area address, and is known as level 2 routing.

If a packet is forwarded via level 2 routing, eventually it will reach the destination area. There, it is handed off to a level 1 router for delivery to the actual node. Segmentation of routing into a hierarchy permits large networks to be built. Level 1 routers exchange link state packets giving the location of all nodes within an area. Level 2 routers exchange link state packets that show the location of all areas.

The concept of area-based routing is one way that DEC networks can coexist with other OSI-compliant networks. The DEC view of the world is a series of private networks, connected together by public networks (See Figure 2). Within a network, dynamic routing information is exchanged. Across network boundaries, routing information is manual.

Interoperability at the Network Layer is thus achieved in two ways. If a vendor has an ISO 8473 compliant end system, the node can easily become part of a DECnet area. If the node also meets the ISO 9542 ES-IS protocols, than the system will automatically configure itself (using an ES Hello message).



Data packets can cross the boundaries of public and private networks, but routing information is static. Other networks appear as reachable areas in the Level 2 forwarding database.

Figure 2: Interconnection of multiple networks

Instead of making other vendor's equipment part of the DECnet routing domain, the alternative is to keep the networks segmented, and connect the intermediate systems together using a common subnetwork (i.e., X.25) and static routing information.

Transport Layer

At the Transport Layer, DEC has traditionally used the *Network Services Protocol* (NSP), a reliable transport mechanism similar to TCP or the ISO TP4 protocols. In Phase V, DEC now supports three of the TP classes: 0, 2, and 4. However, TP4 is the primary transport protocol for DEC applications; TP0 and TP2 are there simply for interconnection to primitive non-DEC systems.

At this point, the DECnet is a pretty faithful OSI network, with a few additional features for backwards compatibility such as DDCMP and a few supplemental features such as the Dynamically Established Datalink and the Modem Control services.

Session Layer

At the Session Layer, DEC is supporting two different protocols. The OSI Session Layer service is supported, but only for OSI application services. DEC application services will use another Session Layer service, the DEC *Session Control* protocol.

continued on next page

DECnet/OSI Phase V *(continued)*

What we have here is really three kinds of networks: pure OSI, DECnet Phase V, and DECnet Phase IV. The pure OSI network (i.e., FTAM between two hosts) would use OSI Session, TP4, and the relevant lower layer protocols. Communication between Phase IV and Phase V would use the NSP transport, and two interconnected routing domains. The Phase V network would use the DEC Session Control, TP4, and the DEC Phase V routing protocols.

Towers

The number of possible combinations of protocol stacks in such an environment are potentially quite large. To keep track of the path available to a given application, DEC uses a concept of *towers.* A tower is a series of addresses, from the network Layer on up. A tower might thus be DEC routing, TP4, DEC Session Control, and DAP. Another tower might be the same, but with the NSP protocols substituted for TP4. Each node and applications keeps a set of towers, showing the possible combinations of protocols to be used for communication.

To communicate between two nodes and/or applications, the DEC session control layer compares the tower sets and comes up with a common subset that can be used. If there is more than one possible set of towers in common, its up to the initiating node to decide which one is best.

Naming Service

The use of towers is thus one aspect of the DEC Session Control service that is different then the generic OSI service. Another aspect is the integration with the *DNA Naming Service* (DNS). DNS [not to be confused with the Internet Domain Name System] is a distributed, replicated naming service used by the Session Control layer to keep node names and application names, and their corresponding tower sets (addresses).

The integration of DNS allows the user and application to communicate via the network using a logical name. The Session Control layer will translate that into a tower set, which will then be handed down to the appropriate transport layer for initiation of a virtual circuit.

The Naming Service is used in more than just the Session Control layer. It is also an integral part of the *Distributed File Service,* and will presumably form the foundation for any DEC X.500 offerings.

The use of towers and DNS applies only within the DECnet Phase V domain. For interconnection to the outside world, DEC uses the OSI Session Layer. Built on top of that DEC has two major services: FTAM and X.400.

FTAM

DEC's FTAM implementation is fairly robust. It supports the first three document types (unstructured text, sequential string, and unstructured binary). DEC has implemented the recovery functional unit. In addition, DEC has an FTAM/DAP gateway, which allows DECnet nodes to access non-DECnet FTAM systems.

X.400

DEC implements X.400 as a gateway into their own proprietary message-handling system, *MAILbus.* Within a DECnet environment, the *Message Router* is the Message Transfer Agent of MAILbus. It moves messages between user interfaces such as ALL-IN-ONE and gateways. DEC has implemented gateways for Telex, IBM's SNADS and PROFS, TCP/IP SMTP, and X.400. The gateway takes an incoming MAILbus message and, using DEC's Distributed Directory product, translates the address.

It also does any translation of the message body (e.g., ASCII to EBCDIC). The message is then sent into the next message handling system. While DEC doesn't really do X.400 currently, they gateway to that environment, which is equivalent as far as the user is concerned.

Is it OSI?

Is DECnet/OSI OSI? At the lower layers, DEC is indeed using OSI internally. The data link protocols are OSI-compliant, the Network Layer is, and three of the important transport classes have been implemented. At the Session Layer, DEC diverges into two co-existent networks. Pure OSI is used for interconnection to the outside world. DEC protocols are used internally.

The issue is really one of philosophy. Is this a true OSI network if most of the application effort seems to be focused in a DEC proprietary stack? The DEC position is that they are adding value, and that as the OSI standards stabilize and mature, we can expect to see things shift over to the OSI side of the stack.

What about the value added within the strictly DECnet environment? We've looked briefly at the Naming Services. Integration of logical naming into the network has an important implication for a wide variety of different services. DEC uses the name server in the distributed file system, for example, as a way of providing transparent mounting of remote file systems.

In addition to the name service, there are a wide variety of other protocols that DEC is using to distinguish itself from a generic OSI environment. For example, DEC has adopted the Hewlett-Packard/Apollo RPC mechanism and added a multi-thread architecture. DEC also has a distributed time protocol used for synchronizing clocks.

Network Management

DEC also supplements OSI in the area of network management. The management protocols, both DEC and OSI are based on the *Common Management Information Protocol* (CMIP). DEC has supplemented this in two ways. First, DEC has architected the agents and event loggers on an individual node. Different network modules are all able to give events to an event logger, which then uses the CMIP *Management Event Notification* subset to send the events to a variety of event sinks on the network. Likewise, DEC architected the management agent on a node, so incoming management directives are dispatched to the appropriate network modules. These supplemental architectures mean that each module is not forced to implement its own CMIP initiators and responders.

The second area that DEC has architected for network management is the *Management Director.* DEC has done this at two levels. First, there is a *Network Command Language* (NCL) which specifies how a user can input a CMIP directive and how it is parsed into CMIP PDUs. NCL also specifies the way that results are displayed back on the user's terminal or console. NCL is a fairly primitive, command-line based management director. A much more ambitious effort is DEC's *Enterprise Management Architecture* (EMA) and the associated DECmcc (*Management Control Center*) product.

EMA splits the function of the director up into three pieces. The *access module* is responsible for communicating with manageable entities. A CMIP access module would be used for communicating with a DECnet Phase V entity, while a bridge management access module would use a different protocol for managing bridges. In addition to bridges and CMIP, DEC will be supporting T1 multiplexors and the *Local Area Transport* (used for terminal servers).

continued on next page

DECnet/OSI Phase V *(continued)*

Presentation modules are responsible for communicating with user devices. DEC presentation modules include DECwindows and traditional line-oriented interpreters. Finally, there are *function modules.* The point of this architecture is that a function (such as "initialize entity") can be applied over a variety of different network protocols at the same time. It is the responsibility of each of the access modules to communicate with its entities.

TCP/IP

What about TCP/IP? DEC has always given lip service to TCP/IP and UNIX (known as Ultrix in the DEC world). However, if you look at their strategic software offerings (i.e., Rdb or ALL-IN-ONE), those are all in the proprietary environment of DECnet and VMS.

With Phase V, DEC will continue to pursue this strategy. While the marketeers claim that TCP/IP has been "integrated" with DECnet, the reality is that the product is simply bundled in. The only form of integration is through primitive application-layer gateways.

Summary

This article is a very brief introduction to Phase V of DECnet. It hasn't even touched on the *Local Area Transport* protocols (used for X Windows terminals and for terminal servers), or on *VAX clusters* (used for tight integration of multiple VAX systems sharing disk drives). We've barely touched the surface of important topics like EMA, the DEC routing protocols, or the use of protocol towers.

Most importantly, this article doesn't address the all important issues of performance, availability, and cost. The reason is simple: DECnet/OSI doesn't really exist yet! It will be interesting to see if this ambitious architecture from DEC will translate into high-quality, cost-effective products.

Is DECnet/OSI really OSI? *Yes.* It implements key subsets of the OSI architecture, allowing DEC to claim that they are compatible with GOSIP. In addition, it goes significantly beyond OSI in many areas. Some areas, such as the name server, will easily merge with OSI standards (X.500 in this case). The name server provides a lower-level technology onto which X.500 can be easily grafted.

Other areas, such as EMA, are providing things ahead of the OSI committees. EMA addresses OSI network management, but it also addresses the question of a management director in a cross-architectural environment.

Finally, there are a few areas, such as remote data access or terminal access where DEC has its own protocols. The true test of DEC's commitment to open systems will be to observe if the DEC protocols are dropped in favor of OSI protocols as the OSI protocols mature. DEC has made the important architectural changes to allow open systems, but we will have to wait and see what strategic directions the product planners take.

CARL MALAMUD is the author of *DEC Networks and Architectures* (McGraw-Hill, 1989), *INGRES* (Van Nostrand Reinhold, 1989), and most recently, *Analyzing Novell Networks* (Van Nostrand Reinhold, 1990). As a way of atoning for writing a book about a subject so crass as Novell, he is currently writing *Analyzing DECnet/OSI Phase V,* to be published by Van Nostrand Reinhold in 1991. He can be reached as `carl@malamud.com`.

To learn more about DECnet, attend S14; "DECnet in a Multiprotocol Environment," on Wednesday at 3:30pm.

For Further Reading

To learn more about some of the topics covered at INTEROP 90, we recommend the following reading list:

[1] *ConneXions,* Volume 3, No. 3, March 1989: *Special Issue on Network Management.*

[2] *ConneXions,* Volume 3, No. 8, August 1989: *Special Issue on Internetwork Routing.*

[3] *ConneXions*, Volume 4, No. 8, August 1990: *Special Issue on Network Management and Network Security.*

[4] Hobby, R., "The Point-to-Point Protocol (PPP)," *ConneXions,* Volume 4, No. 4, April 1990.

[5] Jolitz, W., "X–Windows: More than Just a Pretty Face," *ConneXions,* Volume 4, No. 5, May 1990.

[6] Dern, D., "Interior Routing Protocols," *ConneXions,* Volume 4, No. 7, July 1990.

[7] Blackshaw, R., "Components of OSI: ISDN," *ConneXions,* Volume 3, No. 4, April 1989.

[8] Korb, J. T., "Standard for the transmission of IP datagrams over public data networks," RFC 877.

[9] ISO 8473-1987, International Organization for Standardization, "Data Communications—Protocols for Providing the Connectionless mode Network Service."

[10] CCITT Blue Book, "Data Communication over the Telephone Network," Series V Recommendations: V.120, 1988.

[11] Cerf, V. & Mills, K., "Explaining the Role of GOSIP," RFC 1169.

[12] Neumann, P., "Long-Term Implications of the Internet Worm," *ConneXions,* Volume 3, No. 4, April 1989.

[13] Ostapik, F., "Effect of the Internet Worm on Security," *ConneXions,* Volume 3, No. 9, September 1989.

[14] Schiller, J., "Kerberos: Network Authentication for Today's Open Networks," *ConneXions,* Volume 4, No. 1, January 1990.

[15] Postel, J., "Book Review: *The Cuckoo's Egg,*" *ConneXions,* Volume 4, No. 1, January 1990.

[16] Dern, D., "Interview with Steve Kent on Internet Security," *ConneXions,* Volume 4, No. 2, February 1990.

[17] Jacobsen, O., "Information Sources," *ConneXions,* Volume 3, No. 12, December 1989.

[18] K. Bowers, T. LaQuey, J. Reynolds, K. Roubicek, M. Stahl, A. Yuan, "FYI on Where to Start—A Bibliography of Internetworking Information," RFC 1175.

We also recommend our continuing series *Components of OSI.* Articles to date include: ISDN, X.400, X.500, The Transport Layer, IS-IS Routing, ES-IS Routing, The Session Service, CLNP, The Presentation Layer, The Application Layer Structure, FTAM, The Security Architecture, and Group Communication. See the *ConneXions* index sheets for details. See also "Coming in Future *ConneXions,*" page 27.

CONNEXIONS
480 San Antonio Road
Suite 100
Mountain View, CA 94040
415-941-3399
FAX: 415-949-1779

CONNEXIONS

EDITOR and PUBLISHER Ole J. Jacobsen

INTEROP®90